

# République Algérienne Démocratique Et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université 20 août 1955 -

Skikda Faculté des Sciences

Département d'Informatique



Mémoire de fin d'études pour l'obtention du diplôme de Master

Académique en informatique.

Option : Génie logiciel avancé et applications (GLAA)

Thème

**Développement d'un système IOT pour la gestion intelligente des absences et le suivi d'engagement des étudiants dans l'université**

Réalisé par :

BOUSSOUF SAMI

Encadré par:

M. Kissoum Yacine

Session : juin 2025

# REMERCIEMENTS

---

**Avant tout, je remercie Dieu qui a éclairé mon chemin et m'a donné la volonté, la santé et la patience nécessaires pour réaliser ce modeste travail.**

**En témoignage de ma gratitude et de mon profond respect, j'adresse mes vifs et sincères remerciements à mon encadreur, Monsieur Kissoum Yacine, pour ses conseils, sa patience, sa disponibilité et son encouragement, qui m'ont permis de mener à bien ce travail dans les meilleures conditions.**

**Je tiens également à remercier les membres du jury qui ont bien voulu consacrer de leur temps et de leurs efforts pour évaluer mon travail.**

**Enfin, j'adresse mes remerciements à tous les professeurs qui m'ont accompagné et enseigné tout au long de ce parcours universitaire.**

## Dédicace

---

**Je dédie ce modeste travail à mes parents, pour leur amour inconditionnel, leur soutien moral et matériel, ainsi que leur compréhension tout au long de ce parcours. Leur présence bienveillante a été une lumière constante dans les moments d'incertitude.**

**À ma sœur, pour son affection, son encouragement et sa présence réconfortante dans les moments difficiles.**

**À mes amis les plus proches, en particulier ceux qui m'ont soutenu moralement lorsque j'étais au plus bas, je vous remercie du fond du cœur pour votre bienveillance et vos paroles qui m'ont redonné force et motivation.**

**À mes camarades d'université, pour les échanges, la camaraderie et les souvenirs partagés qui ont enrichi cette belle aventure universitaire.**

**À toutes ces personnes qui ont cru en moi, ce travail est aussi le vôtre.**

# Résumé

---

Ce mémoire porte sur le développement d'un système intelligent basé sur l'Internet des Objets (IoT) visant à optimiser la gestion des absences et le suivi de l'engagement des étudiants au sein de l'université. En intégrant des technologies telles que les cartes RFID et les systèmes de reconnaissance visuelle, ce système permet d'automatiser la prise de présence et de détecter en temps réel la participation des étudiants pendant les cours.

En favorisant une approche proactive de la gestion pédagogique, ce projet contribue à améliorer la qualité de l'enseignement, à identifier rapidement les cas de désengagement, et à renforcer l'interaction entre étudiants et enseignants. Il représente une avancée significative vers une éducation plus connectée, réactive et centrée sur l'apprenant.

**Mots clés** : Internet des Objets, présence intelligente, suivi d'engagement, RFID, détection de participation.

# Abstract

---

This thesis focuses on the development of an intelligent system based on the Internet of Things (IoT), aiming to optimize the management of student absences and the monitoring of student engagement within the university. By integrating technologies such as RFID cards and visual recognition systems, the system automates attendance tracking and enables real-time detection of student participation during classes.

By promoting a proactive approach to educational management, this project contributes to improving the quality of teaching, quickly identifying cases of disengagement, and strengthening interaction between students and teachers. It represents a significant step towards a more connected, responsive, and learner-centered education.

**Keywords:** Internet of Things, smart attendance, engagement monitoring, RFID, participation detection.

يتناول هذا البحث تطوير نظام ذكي يعتمد على تقنيات إنترنت الأشياء بهدف تحسين إدارة الغيابات وتتبع مدى تفاعل الطلاب داخل الجامعة. ومن خلال دمج تقنيات مثل بطاقات وأنظمة التعرف البصري، يتيح هذا النظام تسجيل الحضور الطلاب أثناء الحصة في الوقت الحقيقي. مشاركة تلقائيًا والكشف عن

من خلال اعتماد نهج استباقي في الإدارة التربوية، يسهم هذا المشروع في تحسين جودة التعليم، والتعرف بسرعة على التفاعل، وتعزيز التواصل بين الطلاب والأساتذة. ويمثل خطوة مهمة نحو تعليم أكثر اتصالاً واستجابة. حالات عدم وتركزًا على المتعلم

**الكلمات المفتاحية:** إنترنت الأشياء، الحضور الذكي، تتبع التفاعل، كشف المشاركة (RFID).

# Table de matières

## REMERCIEMENTS

Dédicace

Résumé

Table de matières

Liste des figures

Liste des tableaux

Introduction générale ..... 1

## Chapitre 1: Contexte de l'étude

1. Introduction..... 3

2. Internet des objets ..... 3

    2.1. Définition d'Internet des Objets ..... 3

    2.2. Objet connecté ..... 4

    2.3. Objet intelligent ..... 4

3. L'évolution d'IOT ..... 4

4. Les composant de L'IOT ..... 6

    4.1. Les capteurs et actionneurs..... 6

    4.2. Les dispositifs embarqués ..... 7

    4.3. La connectivité et les protocoles de communication ..... 8

    4.4. Les plateformes cloud et serveurs ..... 8

5. Les protocoles utilisée dans L'IOT ..... 8

    5.1 MQTT (Message Queuing Telemetry Transport)..... 8

    5.2 CoAP (Constrained Application Protocol) ..... 9

    5.3 HTTP..... 9

6. L'architecture de l'IOT ..... 9

6.1. La couche de perception (Perception Layer).....	10
6.2. La couche réseau (Network Layer).....	10
6.3. La couche de traitement (Processing Layer).....	10
6.4. La couche d'application (Application Layer).....	10
7. Domaine d'applications de l'IOT.....	11
7.1. Appareils portables.....	11
7.2. La santé.....	11
7.3. Surveillance du trafic.....	11
7.4. Agriculture.....	11
7.5. Les Maisons intelligentes.....	12
7.6. Les Ville intelligente.....	12
7.7. Automatisation industrielle.....	12
8.L'importance de l'IOT.....	13
8.1. Automatisation accrue des tâches.....	13
8.2. Amélioration de la sécurité des gens.....	13
8.3. Optimisation des processus industriels.....	13
8.4. Réduction des coûts opérationnels.....	13
8.5. Applications de dispositifs intelligents autonomes.....	13
9. Les défis de l'IOT.....	14
9.1. Vulnérabilités de sécurité.....	14
9.2. Interopérabilité.....	14
9.3. Surcharge d'informations.....	14
9.4. Consommation électrique.....	14
9.5. Complexité.....	14
9.6. Défis de confidentialité.....	14
9.7. Manque de normalisation.....	15
9.8. Compatibilité.....	15

<b>10. Conclusion</b> .....	15
<b>Chapitre 2: IOT dans l'enseignement supérieur</b>	
<b>1. Introduction</b> .....	16
<b>2. L'évolution des technologies éducatives grâce à l'IoT</b> .....	16
<b>3. L'environnement éducatif intelligent</b> .....	17
<b>3.1. L'éducation intelligente</b> .....	18
<b>3.2. Université intelligente</b> .....	18
<b>3.3. Les classes intelligentes</b> .....	18
<b>3.4. Enseignement intelligent</b> .....	19
<b>3.5. Apprentissage intelligent</b> .....	19
<b>3.6 L'évaluation intelligente</b> .....	19
<b>4. Technologies utilisées dans l'université</b> .....	20
<b>5. Applications de l'IoT dans l'enseignement supérieur</b> .....	20
<b>5.1. Suivi automatique de la présence des étudiants</b> .....	20
<b>5.2. Surveillance et sécurité des campus</b> .....	20
<b>5.3. Gestion intelligente des salles et départements</b> .....	21
<b>5.4. Amélioration de l'engagement des étudiants</b> .....	21
<b>5.5. Suivi des équipements</b> .....	21
<b>6. Avantages de l'IoT pour les enseignants et les apprenants</b> .....	21
<b>7. Défis liés à l'intégration de l'IoT dans l'enseignement supérieur :</b> .....	22
<b>8. Description du projet dans le cadre de l'IoT pour l'enseignement supérieur :</b> .....	22
<b>9. Conclusion</b> .....	23
<b>Chapitre 3: Analyse et Conception</b>	
<b>1. Introduction</b> .....	24
<b>2. Choix de conception UML</b> .....	24
<b>3. Les objectives du projet</b> .....	24

<b>4. Les parties prenantes</b> .....	25
<b>5. Les exigences du système</b> .....	25
<b>5.1. Les exigences fonctionnelles</b> .....	25
<b>5.2. Les exigences non fonctionnelles</b> .....	26
<b>6. Conception du système EduTrack</b> .....	27
<b>6.1. Diagramme de contexte</b> .....	27
<b>6.2. Diagramme de cas d'utilisation</b> .....	27
<b>6.2.1. Diagramme de cas d'utilisation du système</b> .....	28
<b>6.2.2. Description textuelle du système</b> .....	29
<b>6.3. Diagramme de séquence</b> .....	31
<b>6.4. Diagramme de classe</b> .....	34
<b>6.5. Diagramme d'activités</b> .....	34
<b>7. Conclusion</b> .....	35
<b>Chapitre 4 : Implementation</b>	
<b>1. Introduction</b> .....	36
<b>2. L'architecture du système</b> .....	36
<b>3. Environnement du travail</b> .....	37
<b>3.1. Langage de développement</b> .....	37
<b>3.1.1. Javascript</b> .....	37
<b>3.1.2. Python</b> .....	38
<b>3.1.3. C/C++ (Arduino)</b> .....	38
<b>3.2. Logiciels utilisés</b> .....	39
<b>3.2.1. Visuel Code Studio</b> .....	39
<b>3.2.2. Arduino IDE</b> .....	40
<b>3.3. Système de Gestion de Base de Données</b> .....	41
<b>3.3.1. MySQL</b> .....	41
<b>3.4. Framework utilisés</b> .....	42

3.4.1. Django .....	42
3.4.2. React.....	42
3.5. Bibliothèques utilisées .....	43
3.5.1. Côté Backend (Django) .....	43
3.5.2. Côté Frontend (React.js).....	43
3.5.3. Côté Caméra (Détection avec Python).....	44
3.5.4. Côté ESP32 .....	44
3.6. Matériels utilisés .....	45
3.6.1. ESP32 .....	45
3.6.2. RFID RC522 .....	45
3.6.3. Écran LCD .....	46
4. Interfaces de l'application.....	47
5. Conclusion .....	53
Conclusion générale .....	54
Bibliographie.....	55

# Liste des figures

Figure 1.1: Représentation graphique des objets connectés dans l'environnement IoT .....	4
Figure 1.2: L'augmentation des objets connectés par rapport le monde humain.....	6
Figure 1.3 : Schéma explicative de capteur et actionneur .....	7
Figure 1.4 : Carte Arduino uno.....	7
Figure 1.5 : Les protocoles de communication dans L'IOT.....	9
Figure 1.6 : L'architecture de L'IOT.....	11
Figure 1.7 : Les domaine d'application de l'IOT.....	12
Figure 2.1 : Croissance du marché mondial de l'IoT dans l'éducation (2020–2028)	17
Figure 2.2 : Les domaine d'application de l'IOT dans l'éducation.....	18
Figure 3.1 : Diagramme de contexte du système .....	27
Figure 3.2 : Diagramme de cas d'utilisation d'étudiant .....	28
La figure 3.3 : Diagramme de cas d'utilisation de l'enseignant.....	29
Figure 3.4 : Diagramme de séquence « Scanner la carte ID » .....	32
Figure 3.5 : Diagramme de séquence « évaluer l'engagement des étudiants ».....	33
Figure 3.6 : Diagramme de classe du système .....	34
Figure 3.7 : Diagramme d'activités du système .....	35
Figure 4.1 : Architecture générale du système EduTrack.....	37
Figure 4.2 : JavaScript logo.....	38
Figure 4.3 : Python logo .....	38
Figure 4.4 : C/C++ logo .....	39
Figure 4.5 : VS Code logo.....	39

Figure 4.6: Interface Visual Studio Code .....	40
Figure 4.7 : Arduino IDE logo.....	40
Figure 4.8 : Interface Arduino IDE.....	41
Figure 4.9 : MySQL logo .....	41
Figure 4.10 : React logo .....	42
Figure 4.11 : ESP32.....	45
Figure 4.12 : RFID RC522 .....	46
Figure 4.13 : Écran LCD 16x2 .....	46
Figure 4.14 : Interface de connexion.....	47
Figure 4.15 : Page d'accueil du système EduTrack.....	47
Figure 4.16 : Interface liste des présences.....	48
Figure 4.17 : Interface de diffusion en direct du vidéo .....	48
Figure 4.18 : Interface liste des participations.....	49
Figure 4.19 : Interface d'évaluation de participation .....	49
Figure 4.20 : Interface de suivi de comportement.....	50
Figure 4.21 : Interface d'évaluation du comportement .....	50
Figure 4.22 : Interface de statistique des étudiants.....	51
Figure 4.23 : Interface d'historique .....	51
Figure 4.24 : Le Système avant scanner la carte .....	52
Figure 4.25 : Le Système Après scanner la carte.....	53

## Liste des tableaux

Table 3.1 : Description textuelle du cas d'utilisation « évaluer l'engagement des étudiants » .....	29
Table 3.2 : Description textuelle du cas d'utilisation « Scanner la carte ID ».....	30

# **Introduction Générale**

## Introduction générale

L'enseignement est l'un des piliers les plus importants du développement individuel et collectif. Il permet aux étudiants d'acquérir des connaissances, de construire leur avenir et de contribuer à l'évolution de la société. L'enseignement supérieur, en particulier, joue un rôle essentiel dans la préparation des jeunes aux réalités du monde professionnel et aux défis de la vie moderne. Pour atteindre ces objectifs, les universités doivent non seulement assurer une qualité d'enseignement élevée, mais aussi offrir un environnement adapté, avec des conditions matérielles et pédagogiques favorables. Elles ont la responsabilité de mettre en place des moyens qui facilitent l'apprentissage et garantissent le bien-être des étudiants, afin de rendre l'expérience universitaire plus efficace et plus équitable.

Cependant, dans le milieu universitaire, certaines tâches comme la gestion des absences et l'évaluation de l'engagement des étudiants restent complexes. L'enregistrement des présences repose souvent sur des méthodes traditionnelles telles que l'appel manuel ou les feuilles d'émargement. Ces méthodes entraînent non seulement une perte de temps importante, notamment dans les classes à effectifs élevés, mais présentent aussi un risque accru de fraude, comme le fait de répondre à la place d'un absent ou de signer pour un camarade. Par ailleurs, l'évaluation de la participation reste largement subjective, les enseignants se basant sur leurs observations personnelles, ce qui peut créer des biais et nuire à l'équité entre étudiants.

Face à ces limites, l'Internet des Objets (IoT) apparaît comme une solution innovante. Cette technologie repose sur l'utilisation d'objets connectés capables de collecter et transmettre des données en temps réel. Dans le domaine de l'enseignement, l'intégration de l'IoT offre de nouvelles perspectives. Elle peut contribuer à améliorer la gestion des établissements, à automatiser la prise de présence, à mieux suivre l'implication des étudiants, et à enrichir l'expérience pédagogique pour les enseignants comme pour les apprenants.

Pour relever ce défi, ce mémoire propose le développement d'une application éducative intelligente nommée EduTrack, visant à améliorer le suivi de la présence et de l'engagement des étudiants. Le système utilise des technologies IoT telles que les cartes RFID et des caméras pour détecter la participation, et propose une visualisation des données en fonction du groupe d'étudiants, et suivi du comportement observé en classe.

Notre plan pour réaliser ce travail est organisé autour quatre chapitres comme suit :

- Chapitre 1 "Contexte de l'étude" ; présente les concepts fondamentaux liés à l'Internet des Objets (IoT), ses composants, son architecture et son importance dans les systèmes modernes.
- Chapitre 2 "L'IoT dans l'enseignement supérieur" ; explore les technologies utilisées dans les établissements universitaires et leur impact sur l'enseignement.

## Introduction générale

---

- Chapitre 3 " Analyse et conception" ; détaille les objectifs du système, les exigences fonctionnelles et non fonctionnelles, à travers une étude conceptuelle approfondie.
- Chapitre 4 "Implémentation" ; décrit les outils de développement utilisés et le processus de mise en œuvre du système proposé.

# **Chapitre 1 : Contexte de l'étude**

## 1. Introduction

L'Internet a complètement transformé notre façon de communiquer et d'accéder à l'information. Il a également donné naissance à de nombreuses technologies de pointe, telles que l'Internet des objets. Cette dernière est utilisée pour connecter les objets au réseau, les collecter et les partager en temps réel pour améliorer divers aspects de nos vies.

Le présent chapitre représente le domaine de l'Internet des objets en abordant plusieurs concepts, notamment sa définition, son évolution, ses composants et son architecture, ainsi que ses domaines d'application et d'autres notions.

## 2. Internet des objets

### 2.1. Définition d'Internet des Objets

Les définitions du terme Internet des Objets (IdO) (en anglais Internet Of Things IoT) sont nombreuses. Cette diversité s'explique surtout par l'abondance des domaines avec lesquels ces objets ont des interactions.

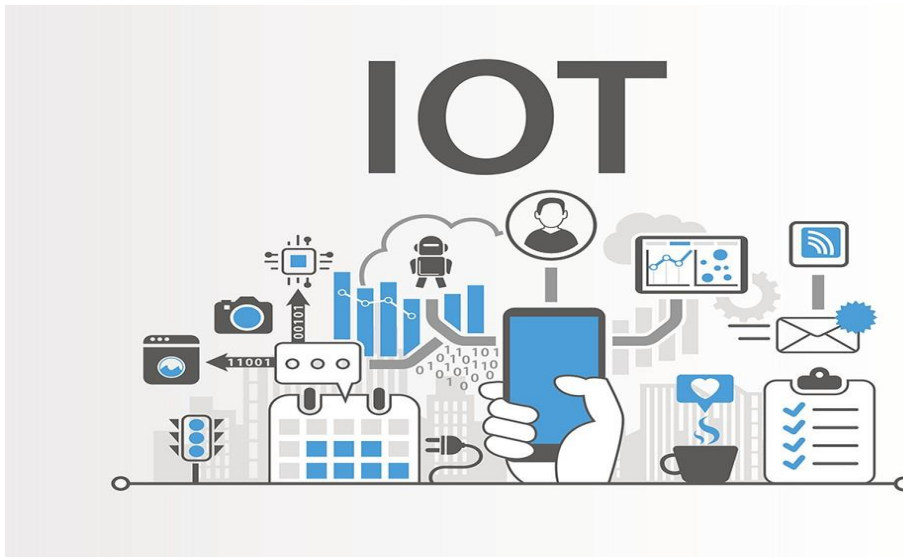
Le groupe de travail Internet of Things Global Standards Initiative (IoT-GSI), piloté par l'International Telecommunication Union (ITU), considère l'IoT comme « une infrastructure mondiale au service de la société de l'information » permettant « d'offrir des services évolués en interconnectant des objets (physiques et virtuels) grâce à l'interopérabilité de technologies de l'information et de la communication existantes ou en évolution ». [1]

De son côté, l'IEEE définit l'IoT comme un « réseau d'éléments chacun muni de capteurs qui sont connectés à Internet ». [1]

Le CERP-IoT « Cluster des projets européens de recherche sur l'Internet des Objets » définit l'Internet des Objets comme : « une infrastructure dynamique d'un réseau global. Ce réseau global a des capacités d'auto-configuration basée sur des standards et des protocoles de communication interopérables. Dans ce réseau, les objets physiques et virtuels ont des identités, des attributs physiques, des personnalités virtuelles et des interfaces intelligentes, et ils sont intégrés au réseau d'une façon transparente ». [2]

En façon générale L'Internet des objets (IoT) fait référence à un ensemble d'objets connectés, qu'ils soient matériels ou virtuels, qui ont la capacité de se connecter et de communiquer les uns avec les autres via Internet. Ces objets, munis de capteurs et de technologies intelligentes, recueillent, partagent et analysent des données afin d'automatiser les services et d'optimiser l'échange avec leur environnement. L'IoT permet donc de concevoir un système intelligent qui facilite la gestion et l'optimisation de diverses tâches quotidiennes et industrielles.

La figure suivante montre une représentation graphique du monde IoT :



**Figure 1.1:** Représentation graphique des objets connectés dans l'environnement IoT

## 2.2. Objet connecté

Un objet ayant l'aptitude d'échanger des données avec d'autres entités qu'elles soient physiques (des objets existent matériellement comme les montres connectées, Équipements industriels ou des drones) ou numériques (des objets peuvent être stockés, traités et récupérés comme les Logiciels et les applications et les Services cloud). [3]

L'idée d'objet connecté (OC) s'étend du simple capteur à l'objet intelligent, c'est-à-dire un objet qui peut effectuer des traitements soit de manière autonome, soit en interaction avec un smartphone ou une plateforme de collecte et de traitement des données IoT. [4]

## 2.3. Objet intelligent

Un objet intelligent est un dispositif matériel équipé de capteurs, de programmes informatiques et d'une connexion Internet, lui permettant de collecter, traiter et transmettre des données avec d'autres appareils et systèmes. Cette capacité à interagir avec son environnement et à adapter son comportement selon les informations reçues est ce qui le rend "intelligent". [5]

## 3. L'évolution d'IOT

L'objet connecté à Internet le plus ancien, remontant à 1982, était un distributeur de boissons installé à l'université Carnegie-Mellon à Pittsburgh, en Pennsylvanie. Cet

appareil avait la capacité de montrer le niveau de remplissage et la température des boissons. [6]

Le terme IoT a été utilisé pour la première fois en 1999 par Kevin Ashton, un ingénieur britannique, pour décrire un système où les objets physiques sont connectés à internet à travers l'utilisation des technologies RFID. [7]

Vers le début des années 2000, des chercheurs et des entreprises ont commencé à explorer les opportunités de relier des appareils et des objets physiques à Internet. Des essais ont été réalisés dans des secteurs comme la domotique, la santé, l'industrie et l'agriculture, Cependant, les technologies requises étaient encore rudimentaires et très coûteuses. [8]

En 2008-2009, le nombre d'objets connectés à Internet dépassait pour la première fois la population mondiale. [6]

Au cours des années 2010, L'évolution des smartphones et des appareils mobiles a facilité l'accès à Internet et favorisé un environnement qui encourage la connexion constante. De même, l'évolution des technologies sans fil comme le Wi-Fi, le Bluetooth a simplifié et rendu plus accessible la communication entre différents objets. Au même temp, le cloud computing a facilité le stockage et l'analyse des grandes quantités de données produites, ouvrant la voie à des applications et analyse plus avancées [8]. Selon Cisco dans son livre blanc, le nombre d'objets connectés en 2010 était de 12,5 milliards d'appareils, soit presque le double de la population humaine. [9]

Aujourd'hui, ce nombre dépasse les 50 milliards d'appareils, une évolution rapide rendue possible grâce à l'intégration de technologies émergentes telles que l'intelligence artificielle, la blockchain et la 5G. [8] [9]

La figure suivante montre l'augmentation des objets connectes par rapport le monde humain :

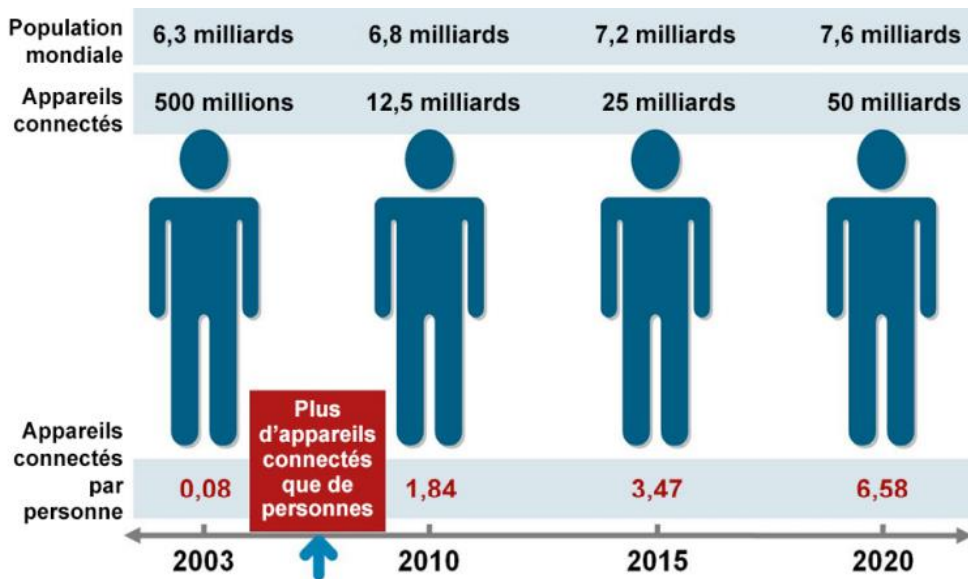


Figure 1.2: L'augmentation des objets connectés par rapport le monde humain. [9]

## 4. Les composant de L'IOT

L'Internet des Objets (IoT) repose sur une infrastructure distribuée et interconnectée, composée de plusieurs éléments techniques qui permettent l'acquisition, la transmission, le traitement et l'exploitation des données issues du monde physique. Ces composants sont généralement organisés en couches fonctionnelles, chacune jouant un rôle clé dans la chaîne de valeur de l'IoT.

### 4.1. Les capteurs et actionneurs

Les capteurs sont les dispositifs physiques de base dans tout système IoT. Ils collectent des données à partir de l'environnement physique : température, humidité, pression, mouvement, luminosité, gaz, etc. Ces données analogiques sont ensuite converties en signaux numériques pour traitement. À l'inverse, les actionneurs permettent d'exécuter une action en réponse à un signal reçu : déclencher une alarme, allumer une LED, ouvrir une vanne, etc. [10]

La figure suivante montre le fonctionnement des capteurs et actionneurs :

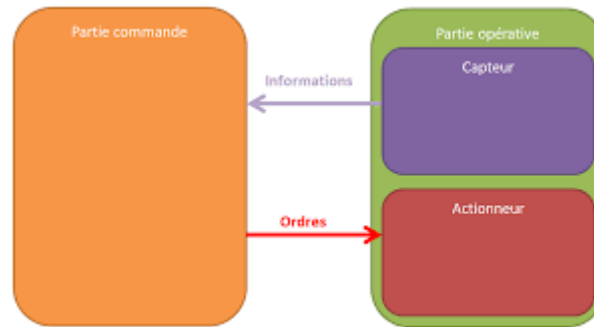


Figure 1.3 : Schéma explicative de capteur et actionneur [11].

## 4.2. Les dispositifs embarqués

Les dispositifs embarqués assurent la collecte, le prétraitement et parfois l'analyse locale des données. Ils intègrent souvent un microcontrôleur ou un processeur. Les plateformes populaires incluent :

- Arduino : simple, idéal pour des projets éducatifs ou de prototypage.
- ESP32 : puissant, avec Wi-Fi et Bluetooth intégrés.
- Raspberry Pi : mini-ordinateur avec plus de puissance de calcul, utilisé pour des applications complexes.

Ces dispositifs sont souvent programmés pour interagir avec les capteurs/actionneurs et envoyer les données via une interface réseau. [12]

La figure suivante montre un exemple d'un dispositif embarqué :



Figure 1 : Carte Arduino uno. [13]

Figure 1.4 : Carte Arduino uno. [13]

### 4.3. La connectivité et les protocoles de communication

La connectivité permet aux objets de transmettre les données à d'autres dispositifs ou à des serveurs distants. Les technologies varient selon le contexte :

- Wi-Fi : utilisé pour des environnements couverts et des débits élevés.
- Bluetooth/BLE : communication courte distance à faible consommation.
- Réseaux cellulaires (4G/5G) : pour des connexions permanentes en mobilité.

Les protocoles comme MQTT, HTTP, ou CoAP assurent l'échange structuré des données sur le réseau. [14]

### 4.4. Les plateformes cloud et serveurs

Les données envoyées par les dispositifs IoT sont souvent centralisées sur des serveurs cloud où elles peuvent être :

- Stockées dans des bases de données.
- Analysées en temps réel à l'aide d'algorithmes (ex. Machine Learning).
- Visualisées via des tableaux de bord ou des applications web.

Des plateformes comme AWS IoT, Google Cloud IoT, ou Microsoft Azure IoT Hub offrent des services adaptés à l'exploitation des données.

## 5. Les protocoles utilisés dans L'IOT

L'Internet des Objets repose sur la communication entre des dispositifs variés souvent limités en puissance, mémoire, ou bande passante. Ainsi, les protocoles de communication utilisés doivent être légers, efficaces et adaptés aux contraintes des environnements embarqués. Parmi les plus répandus, on distingue principalement MQTT, CoAP, http....

### 5.1 MQTT (Message Queuing Telemetry Transport)

Le protocole MQTT est l'un des plus utilisés dans les systèmes IoT en raison de sa légèreté et de son architecture basée sur le modèle publish/subscribe. Les objets connectés (clients) publient des messages à un broker, qui les redistribue aux clients abonnés à un sujet particulier. Il fonctionne sur le protocole TCP et propose trois niveaux de Qualité de Service (QoS) pour garantir la fiabilité des messages.

Ce protocole est particulièrement adapté aux réseaux instables et aux appareils à faible consommation énergétique, notamment dans les domaines de la domotique, de la télémétrie et de la surveillance industrielle. [15]

## 5.2 CoAP (Constrained Application Protocol)

CoAP est un protocole léger conçu spécifiquement pour les objets contraints, fonctionnant sur le protocole UDP. Il adopte un modèle similaire à HTTP (GET, POST, PUT, DELETE), mais avec une surcharge réseau bien inférieure. Il prend en charge le multicast, la fiabilité des messages et un système de découverte des ressources.

Grâce à sa conception optimisée, CoAP est largement utilisé dans les réseaux de capteurs, les systèmes domotiques ou les infrastructures intelligentes à grande échelle. [16]

## 5.3 HTTP

Le protocole HTTP, bien qu'initialement conçu pour le web, est parfois utilisé dans l'IoT, notamment dans des architectures RESTful. Il est simple à mettre en œuvre et compatible avec de nombreuses API Web. Cependant, il est relativement lourd pour les objets à ressources limitées, en raison de la taille de ses en-têtes et de l'usage exclusif de TCP, ce qui en limite l'efficacité énergétique. [10]

La figure suivante montre les protocoles de communication dans L'IOT :

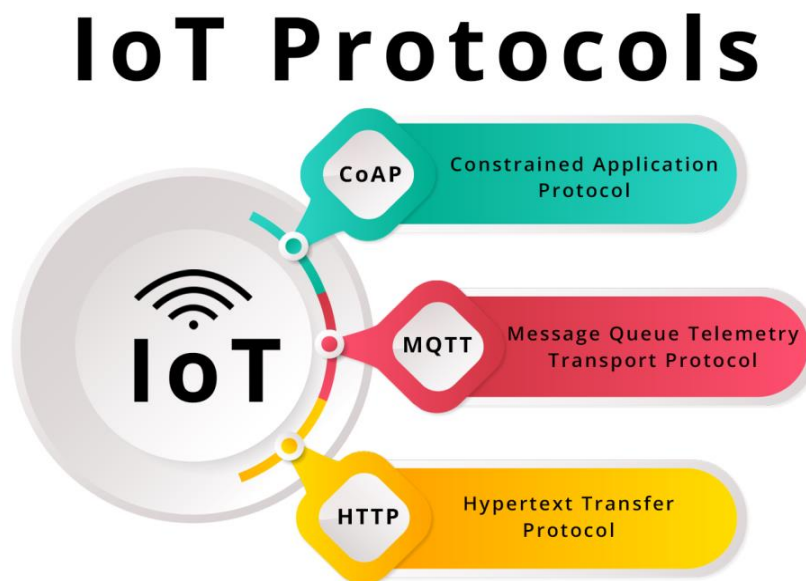


Figure 1.5 : Les protocoles de communication dans L'IOT. [17]

## 6. L'architecture de l'IOT

L'Internet des Objets repose sur une architecture en couches, pensée pour permettre la collecte, la transmission, le traitement et l'exploitation des données issues du monde réel. Cette architecture est généralement organisée en trois à cinq couches fonctionnelles, chacune jouant un rôle spécifique dans la chaîne de valeur de l'IoT.

## 6.1. La couche de perception (Perception Layer)

Appelée aussi couche de détection, elle représente le niveau physique de l'architecture. Elle comprend tous les dispositifs matériels utilisés pour collecter les données du monde réel, comme :

- Les capteurs (température, humidité, mouvement, lumière, etc.)
- Les RFID, caméras, micros, et autres dispositifs de saisie
- Les actionneurs, qui exécutent des commandes (moteurs, relais, etc.)

Cette couche transforme les signaux physiques en données numériques. [10]

## 6.2. La couche réseau (Network Layer)

La couche réseau assure la transmission des données collectées par la couche de perception vers les plateformes de traitement. Elle intègre :

- Les technologies de communication : Wi-Fi, Bluetooth, ZigBee, LoRa, 4G/5G
- Les protocoles réseau (TCP/IP, MQTT, CoAP)
- Les passerelles (gateways) ou nœuds intermédiaires

Elle est responsable du routage, de la sécurité de la transmission, et parfois de la compression des données. [14]

## 6.3. La couche de traitement (Processing Layer)

Cette couche correspond à l'intelligence du système. Elle reçoit les données du réseau, les stocke, les analyse, et prend des décisions automatiques. Les principaux composants sont :

- Les serveurs cloud (Google Cloud, AWS, Azure...)
- Les bases de données (SQL, NoSQL)
- Les algorithmes d'analyse et de traitement (y compris l'IA et le machine learning)

Elle permet de transformer les données brutes en informations utiles.[18]

## 6.4. La couche d'application (Application Layer)

La couche application est celle qui interagit avec l'utilisateur final. Elle fournit des interfaces (applications mobiles, web, tableaux de bord) pour afficher les données, gérer les objets connectés et déclencher des actions.

Exemples d'applications :

- Gestion de présence dans l'éducation
- Santé connectée (monitoring patient)
- Agriculture intelligente
- Surveillance urbaine et sécurité

Elle est spécifique au domaine d'usage et au besoin métier. [12]

La figure suivante montre l'architecture de L'IOT :

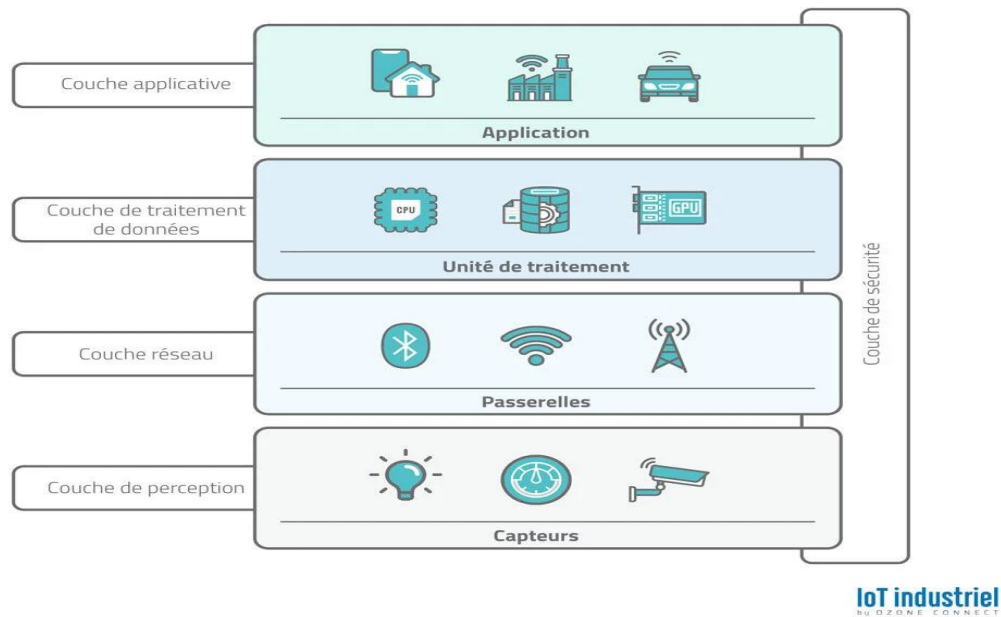


Figure 1.6 : L'architecture de L'IOT. [19]

## 7. Domaine d'applications de l'IOT

### 7.1. Appareils portables

Un des formes les plus couramment rencontrées de l'IoT est la technologie portable, telle que les montres intelligentes, les dispositifs de mesure du glucose et les moniteurs de fréquence cardiaque. Ces appareils intègrent des capteurs qui collectent des informations et, grâce à leur format réduit, ils consomment moins d'énergie [20].

### 7.2. La santé

Dans le domaine des soins médicaux il est utile pour les patients et les médecins. Dans les établissements hospitaliers, des lits intelligents équipés de capteurs sont employés pour surveiller la pression artérielle des patients, la température et d'autres paramètres vitaux [20].

### 7.3. Surveillance du trafic

L'IoT peut aussi être employé pour surveiller le trafic. Nous avons la capacité de surveiller la vitesse du véhicule, d'analyser le flux de circulation, et si une personne ne respecte pas les lois de la route, un ordinateur peut le reconnaître et émettre immédiatement une infraction. [20]

### 7.4. Agriculture

Dans le domaine agricole, l'IoT est utilisé pour évaluer la qualité du sol. Pour surveiller et collecter des données, on installe des capteurs dans les exploitations. On

peut se baser sur les données relatives au sol, telles que l'acidité, les nutriments et l'humidité. Suite à l'analyse de ces informations, il est possible de prévenir les problèmes à court terme et d'améliorer la qualité des cultures. [20]

### 7.5. Les Maisons intelligentes

Chaque composant de la maison, qu'il s'agisse du climatiseur, de la télévision, du réfrigérateur, des ampoules à LED, des ventilateurs, des portes ou encore des machines à laver, a été transformé en équipement intelligent exploitant l'internet des objets. Cela rend la vie bien plus pratique et confortable. [20]

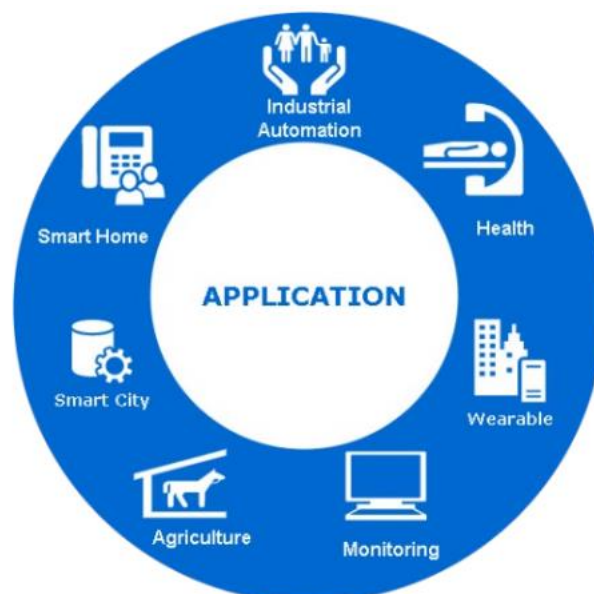
### 7.6. Les Ville intelligente

L'Internet des Objets est exploité dans plusieurs villes pour la gestion du trafic, la gestion des déchets, distribution de l'eau, la gestion de l'énergie électrique et le suivi de la pollution. [20]

### 7.7. Automatisation industrielle

L'automatisation désigne l'exécution d'une tâche sans intervention humaine. La production d'un produit efficace est essentielle pour chaque entreprise, et l'IoT rend cela plus facile. [20]

La figure suivante résume les domaines de L'IOT :



**Figure 1.7** : Les domaine d'application de l'IOT. [20]

## 8.L'importance de l'IOT

L'importance de l'Internet des Objets (IoT) est multiforme et touche divers aspects de la vie quotidienne et des entreprises. Voici quelques points qui soulignent son importance :

### 8.1. Automatisation accrue des tâches

L'IoT offre une automatisation accrue des tâches, rendant ainsi la vie des utilisateurs plus facile. Les dispositifs connectés peuvent être gérés à distance grâce à des applications sur smartphone, ce qui peut être utile et pratique pour les tâches quotidiennes. Comme il a été cité dans le journal 'Idéematic' « Les utilisateurs peuvent allumer et éteindre les lumières, régler la température de leur maison, ou même commander leur nourriture directement depuis leur téléphone portable. ». [21]

### 8.2. Amélioration de la sécurité des gens

Un autre bénéfice de l'intégration de l'IoT dans les applications mobiles est le renforcement de la sécurité. Les appareils connectés peuvent fournir des données en temps réel concernant l'état de leur environnement. Le journal 'Idéematic' a donné un exemple « Tels que la qualité de l'air ou la température, permettant ainsi aux utilisateurs de prendre des mesures pour améliorer leur sécurité et leur santé. ». [21]

### 8.3. Optimisation des processus industriels

L'IoT peut aussi aider à optimiser l'efficacité des procédés industriels, ce qui permet aux entreprises d'économiser énormément. On peut utiliser des capteurs IoT pour contrôler les machines, les équipements et les processus de production, ce qui facilite l'optimisation des opérations de l'entreprise. Un autre exemple a donné par le journal 'Idéematic' a donné un exemple « Par exemple, un fabricant peut utiliser des capteurs IoT pour surveiller les niveaux de stock de matières premières et déclencher automatiquement des commandes de réapprovisionnement pour éviter les pénuries. ». [21]

### 8.4. Réduction des coûts opérationnels

Les solutions IoT ont la capacité d'assister les entreprises dans la diminution de leurs dépenses, contribuant ainsi à leur pérennité financière. Fortinet a cité dans leur livre un exemple « Dans la fabrication, par exemple, les appareils IoT peuvent suivre les équipements et gérer leurs calendriers de maintenance. Les entreprises peuvent économiser de l'argent sur les dépenses CVC grâce à des systèmes automatisés de chauffage et de refroidissement optimisés par l'IoT. ». [22]

### 8.5. Applications de dispositifs intelligents autonomes

Les dispositifs IoT ont gagné en popularité dans plusieurs domaines. Que ce soit dans le secteur médical, du loisir, des transports, de l'hôtellerie ou de l'éducation, l'utilisation de l'IoT a augmenté et continuera d'augmenter à mesure que les utilisateurs

découvriront encore de nouvelles façons d'exploiter des données interactives en temps réel. [22]

## **9. Les défis de l'IOT**

L'IoT s'intègre dans tous les domaines de la vie et implique de nombreuses technologies pour échanger des données entre appareils. Cela rend son développement complexe et conduit à plusieurs défis. Voici quelques points qui soulignent ses défis :

### **9.1. Vulnérabilités de sécurité**

Les appareils IoT recueillent une vaste quantité d'informations quotidiennement, et si ces données ne sont pas correctement sécurisées, ils pourraient être piraté ou manipulés, ce qui Cela pourrait conduire à des infractions de données, du vol d'identité et une perte de la confiance des clients. [20]

### **9.2. Interopérabilité**

Les produits de l'industrie des technologies de l'information reposent sur diverses technologies et protocoles, qui peuvent conduire à des soucis de connexion, des problèmes de compatibilité et des silos de données. [20]

### **9.3. Surcharge d'informations**

L'IdO produit une vaste quantité de données, dont certaines pourraient être inexactes ou non pertinentes si elles ne sont pas correctement traitées. Cela peut aussi entraîner un ralentissement des temps de réponse, une perte de données et potentiellement un plantage du système. [20]

### **9.4. Consommation électrique**

Un grand nombre de dispositifs IoT consomment une quantité importante d'électricité et reposent sur des batteries ou d'autres sources d'énergie, qui restreignent leur utilité. [20]

### **9.5. Complexité**

La conception et la gestion des solutions IoT peuvent être compliquées en raison de leur complexité. Il est nécessaire de former des travailleurs et d'investir de l'argent pour garantir que les composants, les capteurs et les systèmes de communication fonctionnent ensemble parfaitement. [20]

### **9.6. Défis de confidentialité**

Les dispositifs de l'IdO ont la capacité de recueillir des informations personnelles qui, si elles ne sont pas correctement gérées, pourraient mener à des atteintes à la vie privée et à une violation des droits individuels. [20]

### **9.7. Manque de normalisation**

Le Manque de normes dans le secteur de l'IdO peut engendrer des soucis d'interopérabilité, la difficulté à localiser les fournisseurs et un manque de transparence. [20]

### **9.8. Compatibilité**

Avec l'évolution technologique, les équipements plus anciens deviennent incompatibles aux versions les plus récentes, engendrant des difficultés de compatibilité. [20]

## **10. Conclusion**

L'Internet des Objets (IoT) est une technologie innovante qui permet aux objets de communiquer entre eux et de transmettre des données en temps réel. Grâce à ses nombreuses applications, il transforme plusieurs secteurs en rendant les systèmes plus intelligents et efficaces. Malgré certains défis comme la sécurité et la gestion des données, l'IoT reste une solution prometteuse pour bâtir un monde plus connecté et réactif.

Le chapitre suivant concentre sur L'IOT dans L'enseignement supérieur.

## **Chapitre 2: IOT dans l'enseignement supérieur**

## 1. Introduction

L'Internet des Objets (IoT) est une technologie qui permet à des objets connectés, comme des capteurs ou des appareils intelligents, de communiquer entre eux et avec des systèmes informatiques. Dans le domaine de l'enseignement supérieur, l'IoT commence à jouer un rôle important. Grâce à ces objets connectés, il devient possible d'automatiser certaines tâches dans les universités, comme la gestion des présences, le contrôle d'accès aux salles, ou encore le suivi de l'engagement des étudiants. Par exemple, on peut utiliser des cartes RFID pour détecter la présence d'un étudiant, ou des caméras intelligentes pour observer la participation pendant les cours. L'intégration de l'IoT dans les établissements universitaires permet donc d'améliorer l'organisation, de gagner du temps, et de rendre l'apprentissage plus interactif et adapté aux besoins des étudiants.

Dans ce chapitre en va voir l'évolution des technologies éducatives grâce à l'IoT et les technologies utilisées et les applications de l'IoT dans l'enseignement supérieur et d'autres notions.

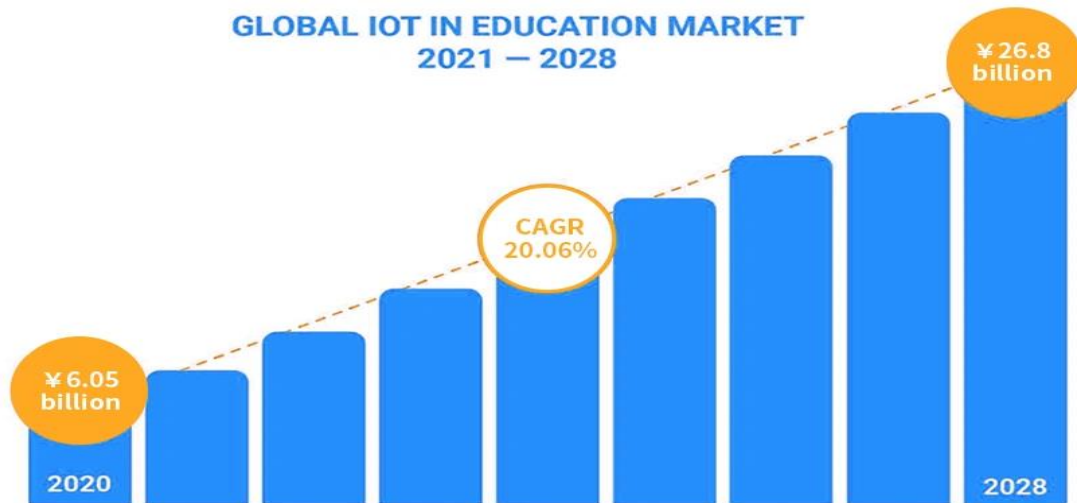
## 2. L'évolution des technologies éducatives grâce à l'IoT

L'Internet des Objets (IoT) a profondément transformé le paysage des technologies éducatives en facilitant la transition d'un enseignement traditionnel vers un apprentissage interactif, dynamique et centré sur l'étudiant. L'IoT permet aujourd'hui de connecter des objets pédagogiques (comme des tableaux interactifs, des capteurs d'environnement, ou des dispositifs de suivi de la participation) afin de créer des salles de classe intelligentes où les conditions d'apprentissage sont continuellement adaptées aux besoins des apprenants. Cette capacité à interagir en temps réel avec l'environnement et à recueillir des données pédagogiques pertinentes constitue un tournant dans les modalités d'enseignement. [23]

De nombreuses recherches récentes ont mis en lumière l'impact positif de l'IoT sur la personnalisation de l'apprentissage. En analysant les comportements des étudiants via des capteurs et dispositifs connectés, les enseignants peuvent désormais ajuster leurs stratégies pédagogiques pour mieux répondre aux rythmes et styles d'apprentissage individuels. Ce processus s'inscrit dans une logique d'éducation adaptative rendue possible par la combinaison de l'IoT, de l'intelligence artificielle et des plateformes d'analyse de données. Il s'agit là d'un aspect fondamental de ce que la littérature désigne comme Éducation 4.0, qui redéfinit les rôles de l'enseignant et de l'apprenant dans un environnement où l'interaction homme-machine devient un levier de progrès académique. [24]

Sur le plan institutionnel, l'IoT permet également d'optimiser la gestion des campus grâce à des systèmes intelligents de contrôle d'accès, de gestion énergétique ou de réservation de ressources. Ces innovations contribuent à l'émergence du campus intelligent (smart campus), concept dans lequel la technologie n'est plus un simple outil de support mais devient un composant structurant de l'écosystème universitaire. [25]

La figure suivante Croissance du marché mondial de l'IoT dans l'éducation (2020–2028) :

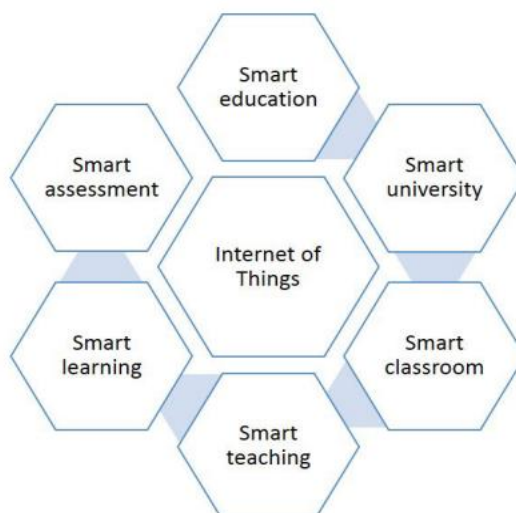


**Figure 2.1** : Croissance du marché mondial de l'IoT dans l'éducation (2020–2028).  
[30]

### 3. L'environnement éducatif intelligent

L'IdO change considérablement tous les aspects de la vie en convertissant chaque objet en un élément intelligent. C'est aussi le cas dans le contexte éducatif où l'on trouve une chaîne véritablement fluide qui transfère l'énergie depuis l'éducation intelligente, de l'université intelligente, de la classe intelligente, de l'enseignement intelligent et de l'apprentissage intelligent à l'évaluation intelligente. [26]

La figure suivante montre les domaines de L'IOT dans l'éducation :



**Figure 2.2** : Les domaines d'application de l'IOT dans l'éducation. [26]

### 3.1. L'éducation intelligente

Des technologies intelligentes comme le cloud computing et l'Internet des Objets contribuent à révolutionner l'éducation en la rendant plus intelligente, tout en jouant un rôle important dans la conception d'un environnement éducatif intelligent. L'objectif de l'éducation intelligente est de doter les employés des compétences et des savoirs du XXI<sup>e</sup> siècle, leur permettant ainsi de faire face aux enjeux sociétaux. L'éducation intelligente est mise en œuvre grâce à une infrastructure IoT, qui se compose de dispositifs de détection, d'applications pour l'utilisateur et de supports de communication. L'usage de l'Internet des Objets dans le secteur éducatif engendrera une amélioration de la qualité du processus pédagogique, puisque les élèves apprendront plus vite et que les enseignants seront capables d'exercer leur activité d'enseignement de façon efficace. [26]

### 3.2. Université intelligente

Une université intelligente incorpore des idées novatrices, associant l'innovation matérielle et logicielle aux salles de classe intelligentes équipées des technologies et processus éducatifs les plus récents. S'appuyant sur des stratégies d'enseignement et d'apprentissage modernes. Une université intelligente fournit un cadre éducatif interactif, l'accès à des ressources internationales et un apprentissage personnalisé basé sur les données recueillies et examinées au sein du réseau.

Actuellement, l'IdO est déployé dans de nombreuses universités, notamment sous forme de caméras, de systèmes de contrôle de la température, d'appareils d'accès aux départements, ainsi que de systèmes électriques et de chauffage. [26]

### 3.3. Les classes intelligentes

Une salle de classe intelligente est un environnement où les activités pédagogiques se déroulent à l'aide d'appareils électroniques tels que les écrans numériques, les

projecteurs vidéo et les appareils connectés à Internet. On le définit comme une salle de classe physique où la gestion, l'accès aux ressources d'apprentissage et l'interaction sont réalisés, combinés à une conscience sensible au contexte. Depuis 2012, les salles de classe intelligentes se fondent sur l'apprentissage mobile, les dispositifs de communication automatisée, les projections vidéo, les caméras, les capteurs, les algorithmes de reconnaissance faciale et d'autres modules surveillant divers paramètres environnementaux. Ces dispositifs intelligents sont reliés via l'Internet des objets, créant ainsi une salle de classe intelligente efficace qui permet de partager des connaissances à tout moment et en tout lieu grâce à un accès communicatif. Les systèmes d'apprentissage basés sur l'IoT peuvent venir en aide aux étudiants souffrant de problèmes de mobilité, d'audition et de vision pour qu'ils puissent étudier à domicile. Cette technologie présente des avantages comme la souplesse, la transmission de savoir, l'amélioration des capacités de réflexion, l'interaction et le partage de contenu pédagogique. [26]

### **3.4. Enseignement intelligent**

L'enseignement intelligent se distingue de l'éducation classique, notamment par la façon dont le contenu est transmis via divers équipements électroniques. En outre, le contenu est accessible en tout temps et l'apprentissage s'adapte à l'utilisateur. L'IdO, via ses systèmes de détection, a la capacité de fournir une liaison avec le monde réel. Cela révolutionne les méthodes pédagogiques en les transformant en une expérience enrichissante. Ces approches d'enseignement doivent être ajustées pour s'adresser aux étudiants confrontés à des obstacles d'apprentissage (troubles visuels, auditifs ou de mobilité). [26]

### **3.5. Apprentissage intelligent**

L'apprentissage intelligent est un processus d'apprentissage adaptatif pour les appareils électroniques. Le processus d'apprentissage intelligent est caractérisé par une étude centrée sur les étudiants et le contenu. Il accorde moins d'importance aux dispositifs et à leur performance, son adaptabilité et son efficacité découlent de la structure des technologies de l'information et de la communication. Les applications d'éducation en ligne fondées sur l'IdO sont essentielles, particulièrement pour l'établissement d'une salle de classe virtuelle et la création d'un environnement d'étude compétitif à l'échelle locale et internationale. L'IdO promeut aussi l'apprentissage en ligne autonome, permettant aux étudiants de se connecter à divers laboratoires ou bibliothèques à travers le monde pour s'engager dans des expériences, rassembler des informations, recevoir des tâches et transmettre leurs évaluations personnelles. [26]

### **3.6 L'évaluation intelligente**

L'évaluation intelligente progresse vers un processus inévitable fondé sur les TIC et se développe dans le cadre de l'écosystème IoT. Les systèmes d'enseignement modernes doivent incorporer des dispositifs et appareils IoT pour évaluer la concentration d'un étudiant, aspect essentiel dans l'évaluation de l'éducation. L'éducation intelligente suppose aussi l'émergence de nouvelles formes

d'enseignement et d'apprentissage, suggérant des approches et des facteurs à considérer dans les tactiques d'évaluation. L'évaluation intelligente devrait inclure des outils et techniques pour identifier la fraude, le plagiat et les situations où les étudiants mémorisent les réponses aux devoirs. Les simulations utilisées lors de l'enseignement constituent un aspect important de l'évaluation intelligente et une technique d'apprentissage potentielle. [26]

## 4. Technologies utilisées dans l'université

Les technologies constituent un outil efficace pour impliquer les étudiants tout en respectant leur niveau de confort. Les institutions éducatives exploitent déjà de plus en plus les technologies (de manière directe et indirecte pour accompagner les étudiants). Par exemple, voici une petite sélection des technologies actuelles dans le secteur de l'éducation. [27] :

- Caméras et vidéo
- Suivi des autobus (transport)
- Cartes d'étudiant (RFID)
- Diffusion et téléviseurs intelligents
- Imprimantes 3D
- Éclairage électrique
- Suivi des présences
- Serrures de porte sans fil
- Capteurs de température
- ....

## 5. Applications de l'loT dans l'enseignement supérieur

### 5.1. Suivi automatique de la présence des étudiants

L'une des premières applications concrètes de l'loT dans les universités est le suivi automatique de la présence des étudiants, souvent mis en œuvre via des cartes RFID ou des balises Bluetooth. Ce système permet de détecter automatiquement l'entrée des étudiants dans les salles de cours et d'enregistrer leur présence sans intervention humaine, réduisant ainsi les erreurs et les oublis. [28]

### 5.2. Surveillance et sécurité des campus

La surveillance et la sécurité des campus s'appuient également sur l'loT à travers des réseaux de capteurs (mouvement, fumée, intrusion) et des caméras connectées en temps réel. Ces dispositifs renforcent la prévention des incidents et la réactivité des équipes de sécurité. Des études ont montré que l'intégration de solutions IoT dans les systèmes de vidéosurveillance permettait une détection proactive des comportements suspects, contribuant à la sécurité globale des environnements universitaires. [29]

### **5.3. Gestion intelligente des salles et départements**

En matière de gestion intelligente des infrastructures, les capteurs IoT régulent automatiquement la lumière, la température et la ventilation selon l'occupation des salles. Cette gestion adaptative contribue à la fois au confort des utilisateurs et à la réduction de la consommation énergétique. [25]

### **5.4. Amélioration de l'engagement des étudiants**

L'IoT est aussi utilisé pour améliorer l'engagement des étudiants. En analysant des signaux captés via des caméras, des microphones ou des dispositifs portables (bracelets connectés), il est possible de mesurer l'attention, les interactions, voire la participation gestuelle des étudiants. Ces données permettent à l'enseignant d'adapter le rythme ou la méthode pédagogique. [30]

### **5.5. Suivi des équipements**

L'inventaire connecté constitue une application logistique essentielle dans les laboratoires et bibliothèques. Les objets (matériel, ressources pédagogiques, livres) sont étiquetés avec des puces RFID ou des QR codes, facilitant leur localisation, leur prêt ou leur maintenance. Cela permet une gestion en temps réel, évite les pertes, et réduit le temps de recherche. [14]

## **6. Avantages de l'IoT pour les enseignants et les apprenants**

L'Internet des Objets (IoT) offre aux enseignants un accès en temps réel à des données pédagogiques, permettant une meilleure compréhension du comportement et du progrès des étudiants. Les tableaux de bord connectés fournissent des indicateurs sur la participation, l'attention et l'assiduité, facilitant ainsi la prise de décision pédagogique rapide et ciblée. Cette approche fondée sur les données soutient la transition vers une pédagogie différenciée et réactive. [23]

Pour les apprenants, l'IoT favorise un apprentissage plus personnalisé et immersif. Grâce aux objets connectés (bracelets, capteurs de posture, dispositifs de réponse en classe), les étudiants reçoivent des retours instantanés et peuvent interagir de manière plus engageante avec le contenu. Cela améliore la motivation, la mémorisation et le sentiment d'autonomie. Une étude de Fernández-Batanero et al. (2022) a montré que les étudiants exposés à des environnements IoT avaient une meilleure perception de leur apprentissage, notamment en termes de clarté, de rythme et d'adaptation. [24]

Du côté organisationnel, l'IoT allège les charges administratives des enseignants en automatisant des tâches répétitives comme l'enregistrement des présences ou la gestion des ressources. Cela leur permet de consacrer davantage de temps à la conception pédagogique et à l'accompagnement individuel. Ces bénéfices ont été particulièrement observés dans les universités ayant intégré des plateformes IoT centralisées.

## 7. Défis liés à l'intégration de l'IoT dans l'enseignement supérieur :

L'un des défis majeurs de l'intégration de l'IoT dans les établissements éducatifs concerne la sécurité et la protection des données. Les dispositifs IoT génèrent de grandes quantités de données sensibles (présence, comportement, localisation), ce qui les rend vulnérables aux cyberattaques ou aux fuites d'informations. Cependant, de nombreux établissements ne disposent pas encore d'infrastructures informatiques suffisamment robustes ni de protocoles de sécurité adaptés à l'IoT. [24]

Un second défi concerne le coût élevé de mise en œuvre des solutions IoT. L'installation de capteurs, d'infrastructures réseau, de plateformes de gestion des données et d'interfaces utilisateurs requiert des investissements initiaux considérables. Ces coûts comprennent également la maintenance, le renouvellement des équipements et l'actualisation des logiciels. Les universités des pays en développement rencontrent souvent des difficultés à financer ces technologies, limitant ainsi leur adoption à grande échelle. [31]

Enfin, la formation du personnel enseignant et administratif constitue un frein important à l'efficacité des projets IoT. L'usage optimal des technologies connectées exige des compétences spécifiques en analyse de données, en cybersécurité et en pédagogie numérique. Toutefois, de nombreux enseignants ne sont pas formés à ces outils, ce qui crée un écart entre les potentiels techniques offerts par l'IoT et leur exploitation effective en classe. [32]

## 8. Description du projet dans le cadre de l'IoT pour l'enseignement supérieur :

Le projet s'inscrit dans la dynamique de transformation numérique des établissements d'enseignement supérieur grâce à l'Internet des Objets (IoT). Il vise à développer un système intelligent de gestion des présences et de suivi de l'engagement des étudiants, combinant plusieurs technologies IoT telles que les lecteurs RFID et la vision par ordinateur.

Concrètement, chaque étudiant est muni d'une carte RFID unique, qui est scannée à l'entrée de la salle via un module ESP32 connecté à un lecteur RFID et à un écran LCD. L'ESP32 se charge d'envoyer le UID de la carte au serveur via WiFi. Ce serveur, développé avec Django, vérifie l'identité de l'étudiant et enregistre sa présence dans une base de données MySQL. Parallèlement, une caméra connectée utilise des algorithmes de reconnaissance faciale et de détection de gestes pour identifier les étudiants qui participent activement au cours (ex. : lever la main).

Une interface web développée avec React permet aux enseignants de :

- Visualiser en temps réel la liste des étudiants présents,

- Évaluer les participations (bonus/malus),
- Consulter l'historique des présences et des interactions,
- Générer des statistiques quotidiennes sur l'engagement en classe.

Ce système offre une solution complète et automatisée, réduisant les tâches administratives, fiabilisant les données de présence, et permettant une évaluation plus fine de l'implication des étudiants. Il répond aux enjeux de modernisation des pratiques pédagogiques à travers une approche intégrée de l'IoT dans le contexte universitaire.

## 9. Conclusion

L'intégration de l'Internet des Objets (IoT) dans l'enseignement supérieur représente une opportunité précieuse pour moderniser les pratiques pédagogiques et améliorer la gestion des établissements. En automatisant des tâches comme la prise de présence, en facilitant le suivi de l'engagement des étudiants et en optimisant l'utilisation des ressources, l'IoT contribue à créer un environnement d'apprentissage plus interactif, efficace et adapté aux besoins des apprenants.

Dans le chapitre suivant en va faire l'analyse et la conception pour notre système proposé Edutrack.

## **Chapitre 3: Analyse et Conception**

## 1. Introduction

La réalisation d'un projet informatique nécessite l'utilisation d'une méthode de conception, qui est l'une des phases essentielles du processus de développement de projets. Cette méthode vise à analyser le problème et à concevoir des solutions efficaces et simples à mettre en œuvre.

Ce chapitre présente notre choix de modélisation, qui est l'UML. Nous y exposons une analyse du système, en précisant les objectifs ainsi que les parties prenantes et les exigences. Nous exploitons également quelques diagrammes pour modéliser le système proposé.

## 2. Choix de conception UML

Notre projet EduTrack repose sur une architecture client-serveur moderne, où les rôles du client et du serveur sont bien distincts. Le client est une application web développée sous forme de SPA (Single-Page Application) à l'aide de React, ce qui permet une expérience utilisateur fluide, sans rechargement de page. Toutes les interactions avec le serveur se font via des requêtes HTTP, permettant au client d'envoyer ou de récupérer les données à la demande.

Le serveur, basé sur le framework Django, agit en tant que backend central. Il expose une API RESTful qui traite les requêtes du client et répond avec des données structurées (généralement en format JSON). Cette architecture favorise la modularité, la scalabilité, ainsi que l'indépendance entre l'interface utilisateur et la logique métier, tout en offrant une base robuste pour intégrer des dispositifs IoT. C'est notamment le cas du module ESP32, qui joue le rôle d'un client physique connecté : il lit les cartes RFID des étudiants, envoie l'identifiant unique (UID) au serveur via le WiFi, et affiche la réponse (comme le nom de l'étudiant) sur un écran LCD. Cette intégration renforce l'automatisation du système de pointage et illustre la compatibilité du backend avec des capteurs connectés en temps réel.

## 3. Les objectives du projet

L'objectif principal de ce projet est de développer un système intelligent et connecté capable d'automatiser la gestion de la présence et de la participation des étudiants dans un environnement universitaire. Ce système vise à améliorer l'efficacité du pointage des présences, à minimiser les fraudes, et à faciliter le travail des enseignants pour faciliter les méthodes d'évaluations en fournissant des outils numériques modernes. Pour cela, plusieurs objectifs précis ont été définis :

- Automatisation de la prise de présence : Grâce à l'utilisation de cartes RFID individuelles, chaque étudiant pourra enregistrer sa présence en approchant simplement sa carte d'un lecteur connecté à un microcontrôleur Esp32 au

- moment de son entrée. Ce système élimine le besoin de listes papier et réduit considérablement le temps nécessaire pour faire l'appel.
- Transmission en temps réel des données : Les informations lues par le système RFID seront transmises automatiquement vers une base de données. Cela permet un enregistrement immédiat et sécurisé des données, et accessibles aux enseignants pour vérifier la liste des étudiants présents.
  - Détection automatique de la participation Le système utilise une caméra pour assurer deux fonctions essentielles : l'identification des étudiants par reconnaissance faciale, et la détection de la participation par la reconnaissance du geste de la main levée. Grâce à des bibliothèques de vision par ordinateur comme OpenCV et des modèles d'IA adaptés, l'application sera capable d'identifier automatiquement les visages des étudiants présents, et de détecter ceux qui interagissent activement pendant le cours.
  - Mise à disposition d'une interface de gestion : Une interface sera accessible aux enseignants via un logiciel. Elle permettra d'afficher les présences, de suivre l'évolution de la participation ou encore suivre le comportement des étudiants.

## 4. Les parties prenantes

Les parties prenantes sont toutes les personnes ou systèmes concernés par le fonctionnement, l'utilisation ou le développement du système.

Les enseignants : Ils suivent l'engagement et le comportement des étudiants en vérifiant leur présence et en ajoutant des évaluations.

Les étudiants : Ils assistent aux cours, font des efforts et s'engagent activement afin d'être évalués.

Les développeurs : Ils conçoivent, développent le système et assurent son bon fonctionnement.

La plateforme Progress : Système externe utilisé pour la récupération des notes des étudiants. Elle interagit avec le système via une API.

## 5. Les exigences du système

### 5.1. Les exigences fonctionnelles

Les exigences fonctionnelles décrivent les services que le système doit fournir et les comportements qu'il doit adopter face à certaines entrées ou situations. Elles définissent les fonctionnalités attendues du système du point de vue de l'utilisateur final. Voici les fonctionnalités que le système doit faire :

Authentification des utilisateurs : Le système doit permettre aux enseignants de se connecter via un nom d'utilisateur et un mot de passe.

Enregistrement des présences via RFID : Lorsqu'un étudiant approche sa carte RFID du lecteur, le système doit identifier l'étudiant et enregistrer sa présence dans la base de données.

Affichage des informations sur écran LCD : Après lecture de la carte RFID, les informations de l'étudiant (nom, prénom, groupe...) doivent s'afficher sur un écran LCD.

Envoi des données vers le serveur : Les données de présence doivent être automatiquement envoyées vers une base de données centrale.

Consultation des présences via une interface web : Les enseignants doivent pouvoir consulter les présences enregistrées via une interface web sécurisée, qui indique si l'étudiant est arrivé à l'heure ou en retard.

Reconnaissance faciale : Le système doit être capable de reconnaître automatiquement les étudiants à l'aide de la reconnaissance faciale pour aide à l'évaluation.

Reconnaissance des gestes : Le système doit détecter automatiquement les gestes de participation des étudiants (lever la main) à l'aide d'une caméra connectée.

Affichage des participations : Le système doit afficher la liste des étudiants ayant levé la main pendant le cours.

Évaluation de l'engagement : Le système doit permettre aux enseignants d'évaluer la participation des étudiants pendant les cours et d'ajouter des remarques ou des points bonus.

Évaluation de comportement : Le système doit permettre aux enseignants d'évaluer le comportement des étudiants pendant la séance et d'attribuer des bonus ou des malus.

Affichage de la vidéo : Le système doit permettre l'affichage en direct du vidéo capturé par la caméra.

Afficher des statistiques sur chaque étudiant : Le système doit afficher des graphes visuels qui montre les statistiques de chaque étudiant.

Recherche des données : Le système doit permettre une recherche rapide des données (par nom, date) dans l'historique ou dans la liste de présence.

Récupération des notes des étudiants : Le système doit permettre la récupération des notes des étudiants depuis le système de gestion académique PROGRESS.

## **5.2. Les exigences non fonctionnelles**

Les exigences non fonctionnelles définissent les qualités générales que le système doit respecter. Voici les exigences non fonctionnelles que le système doit respecter :

Fiabilité : Le système doit fonctionner de manière continue pendant les cours, sans interruption ni perte de données.

Performance : La lecture d'une carte RFID et l'enregistrement de la présence doivent s'effectuer en quelques secondes.

Sécurité : L'accès aux données doit être protégé par des mécanismes d'authentification.

Utilisabilité : L'interface web doit être simple, claire et facile à utiliser pour les enseignants, même sans formation technique.

## 6. Conception du système EduTrack

### 6.1. Diagramme de contexte

Le diagramme de contexte représente une vue globale du système et illustre ses interactions principales avec les parties externes (acteurs). Il permet de comprendre rapidement les entrées, les sorties et les flux d'information entre le système et son environnement. La figure ci-dessous représente le diagramme de contexte de notre système :

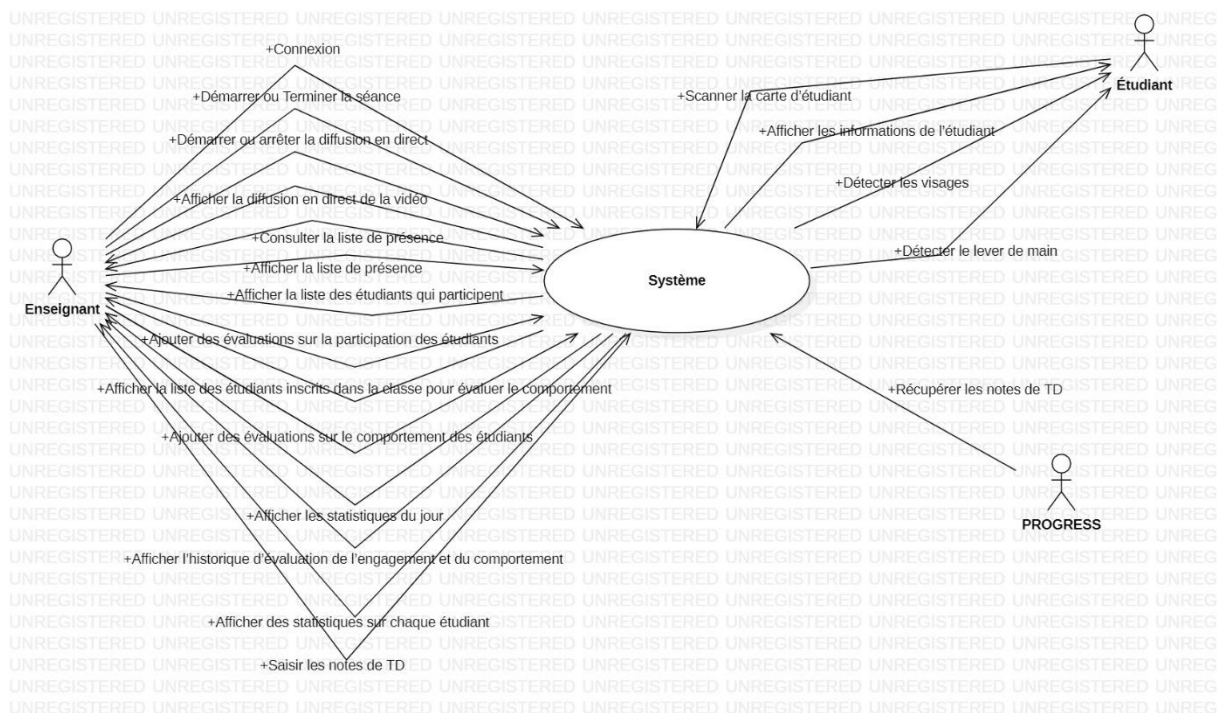


Figure 3.1 : Diagramme de contexte du système.

### 6.2. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est un outil de modélisation UML qui permet de représenter les fonctionnalités principales du système du point de vue des utilisateurs

(ou "acteurs"). Il illustre les interactions entre les utilisateurs et le système, en mettant en évidence les différents cas d'usage possibles.

### 6.2.1. Diagramme de cas d'utilisation du système

La figure 2.2 représente le diagramme de cas d'utilisation de l'étudiant :

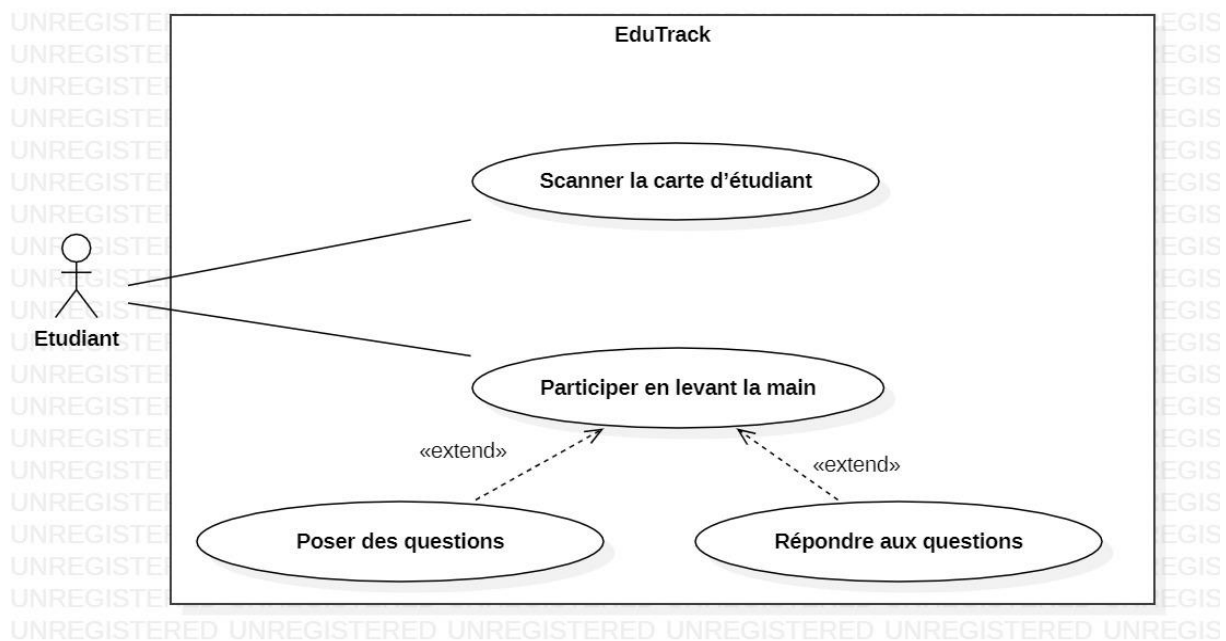


Figure 3.2 : Diagramme de cas d'utilisation d'étudiant.

La figure 2.3 représente le diagramme de cas d'utilisation de l'enseignant :



La figure 3.3 : Diagramme de cas d'utilisation de l'enseignant.

### 6.2.2. Description textuelle du système

**Cas d'utilisation :** évaluer l'engagement des étudiants

**Acteur :** Enseignant

**Résumé :** Ce cas d'utilisation permet à l'enseignant d'évaluer et d'enregistrer la participation des étudiants pendant une séance de cours.

**Préconditions:**

- L'enseignant est authentifié dans le système.
- L'enseignant démarré la séance.
- Les données d'interaction (mains levée) sont disponibles.

**Postconditions:**

- L'évaluation est enregistrée dans l'historique du système.
- Ces données sont disponibles pour consultation.

**Scénario nominal:**

- L'enseignant accède à l'interface d'évaluation de participation.
- Le système affiche la liste des étudiants détectés après leur engagement (ex. : main levée, questions posées).
- L'enseignant sélectionne un étudiant pour évaluer.
- L'enseignant choisit une évaluation adaptée :
  - Choisir l'état de participation (bonne réponse, fausse réponse, pose une question).
  - Ajouter une remarque.
  - Peut ajouter des points bonus.
- Le système enregistre les données d'évaluation.

**Table 3.1 :** Description textuelle du cas d'utilisation « évaluer l'engagement des étudiants ».

**Cas d'utilisation :** Scanner la carte ID

**Acteur :** Étudiant

**Résumé :** Ce cas d'utilisation permet de permettre à l'utilisateur d'identifier automatiquement un étudiant au début d'une séance à l'aide de sa carte RFID, pour enregistrer sa présence.

**Préconditions :**

- L'étudiant possède une carte étudiant valide.
- Le dispositif de scan (lecteur de carte RFID) est opérationnel et connecté au système.

**Postconditions :**

- L'étudiant est marqué comme présent pour la session.
- Son identité est liée à toute autre interaction ultérieure (participation, comportement).

**Scénario nominal:**

- L'étudiant approche sa carte ID du lecteur.
- Le système lit l'identifiant unique de la carte.
- Si la carte est valide :
  - L'identité de l'étudiant est confirmée.
  - La présence est automatiquement enregistrée pour la session dans la base de données et sera affichée dans l'interface utilisateur.

- Le système affiche un message sur un écran qui contient les informations de l'étudiant.

**Scénario alternatif :**

- Si la carte est invalide : un message d'erreur est affiché et la présence n'est pas enregistrée.

**Table 3.2 :** Description textuelle du cas d'utilisation « Scanner la carte ID ».

### 6.3. Diagramme de séquence

Le diagramme de séquence est un diagramme UML dynamique qui permet de modéliser les interactions entre les différents objets ou acteurs du système au fil du temps. Il illustre comment les messages sont échangés pour accomplir un scénario spécifique, en mettant en évidence l'ordre chronologique des actions. La figure ci-dessous représente le diagramme séquence de notre système :

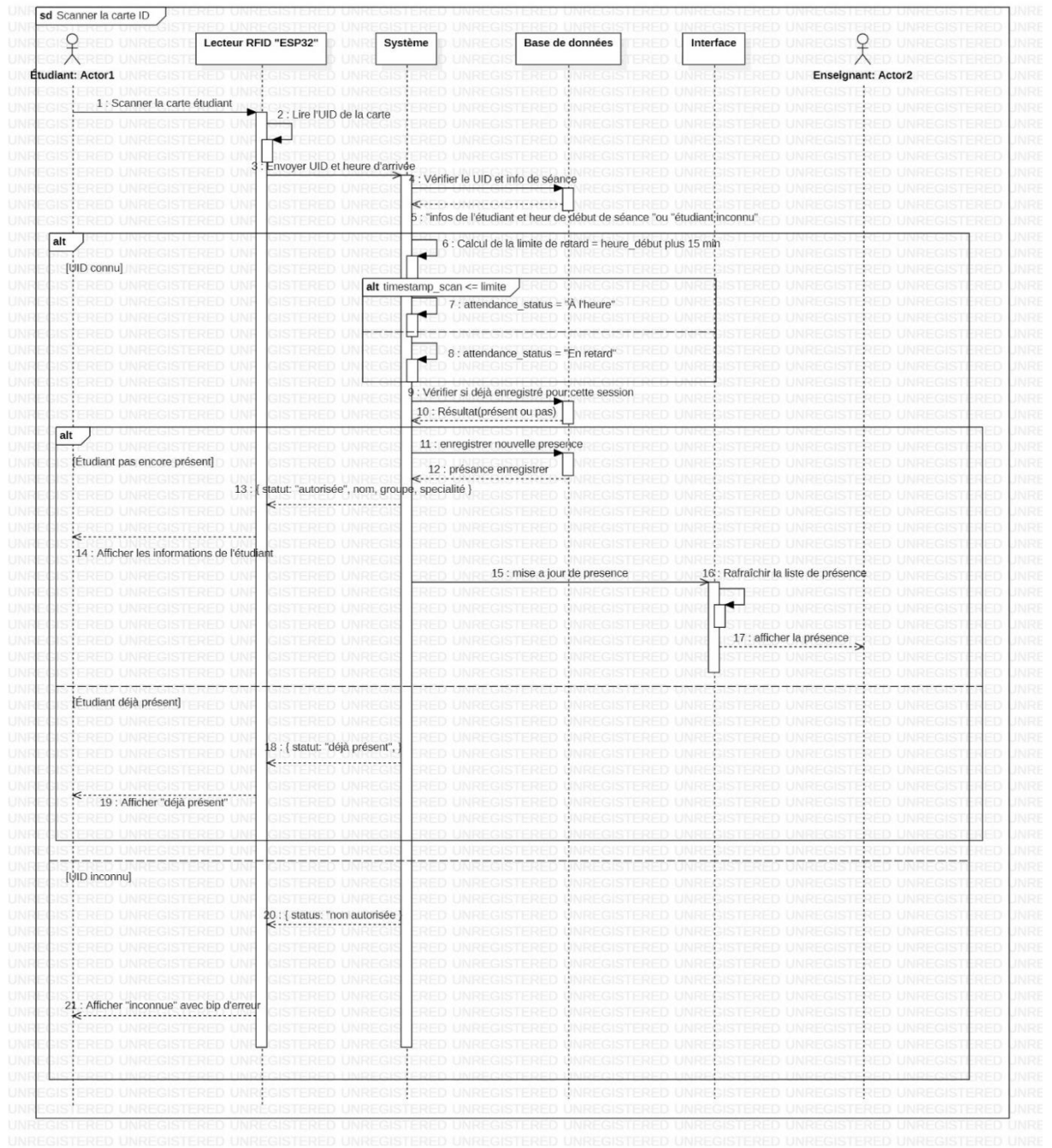


Figure 3.4 : Diagramme de séquence « Scanner la carte ID ».

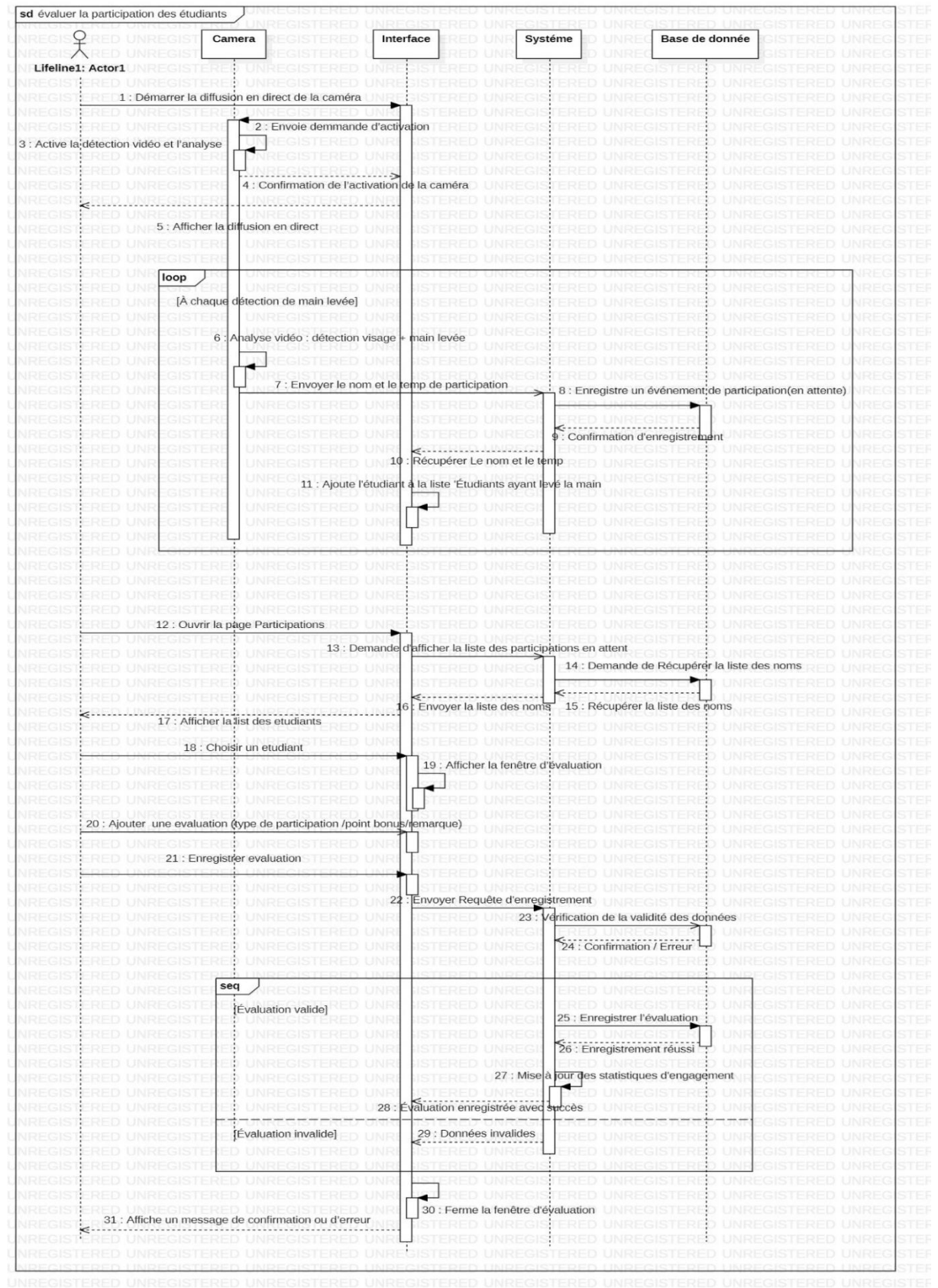


Figure 3.5 : Diagramme de séquence « évaluer l'engagement des étudiants ».

### 6.4. Diagramme de classe

Le diagramme de classes est un élément fondamental de la modélisation UML. Il permet de représenter la structure statique du système, en définissant les classes principales, leurs attributs, leurs méthodes (opérations), ainsi que les relations entre elles (association, héritage, composition, etc.). La figure ci-dessous représente le diagramme de notre système :

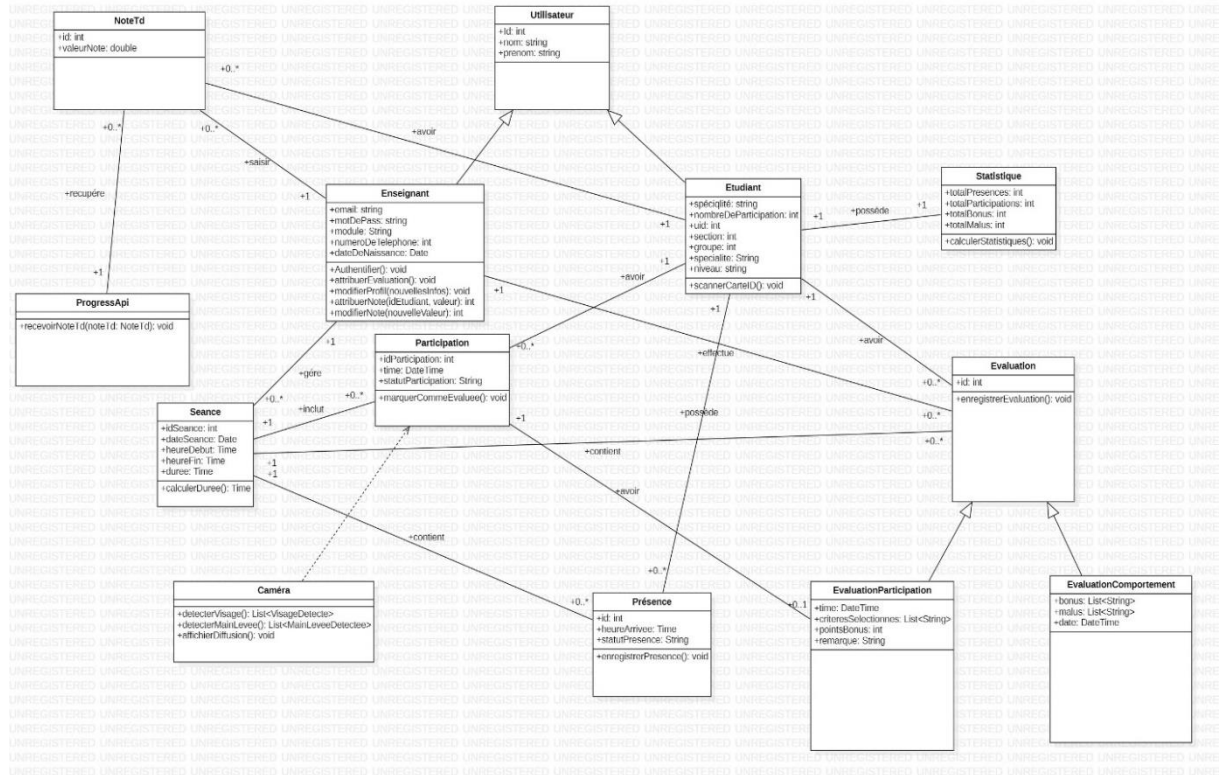


Figure 3.6 : Diagramme de classe du système.

### 6.5. Diagramme d'activités

Le diagramme d'activités est un diagramme UML comportemental qui permet de modéliser les flux de travail (workflow) ou les enchaînements d'actions dans un processus métier ou une fonctionnalité du système. Il est particulièrement utile pour décrire le déroulement logique d'un scénario, incluant les décisions, les alternatives, et les parallélismes. La figure ci-dessous représente le diagramme d'activités de notre système :

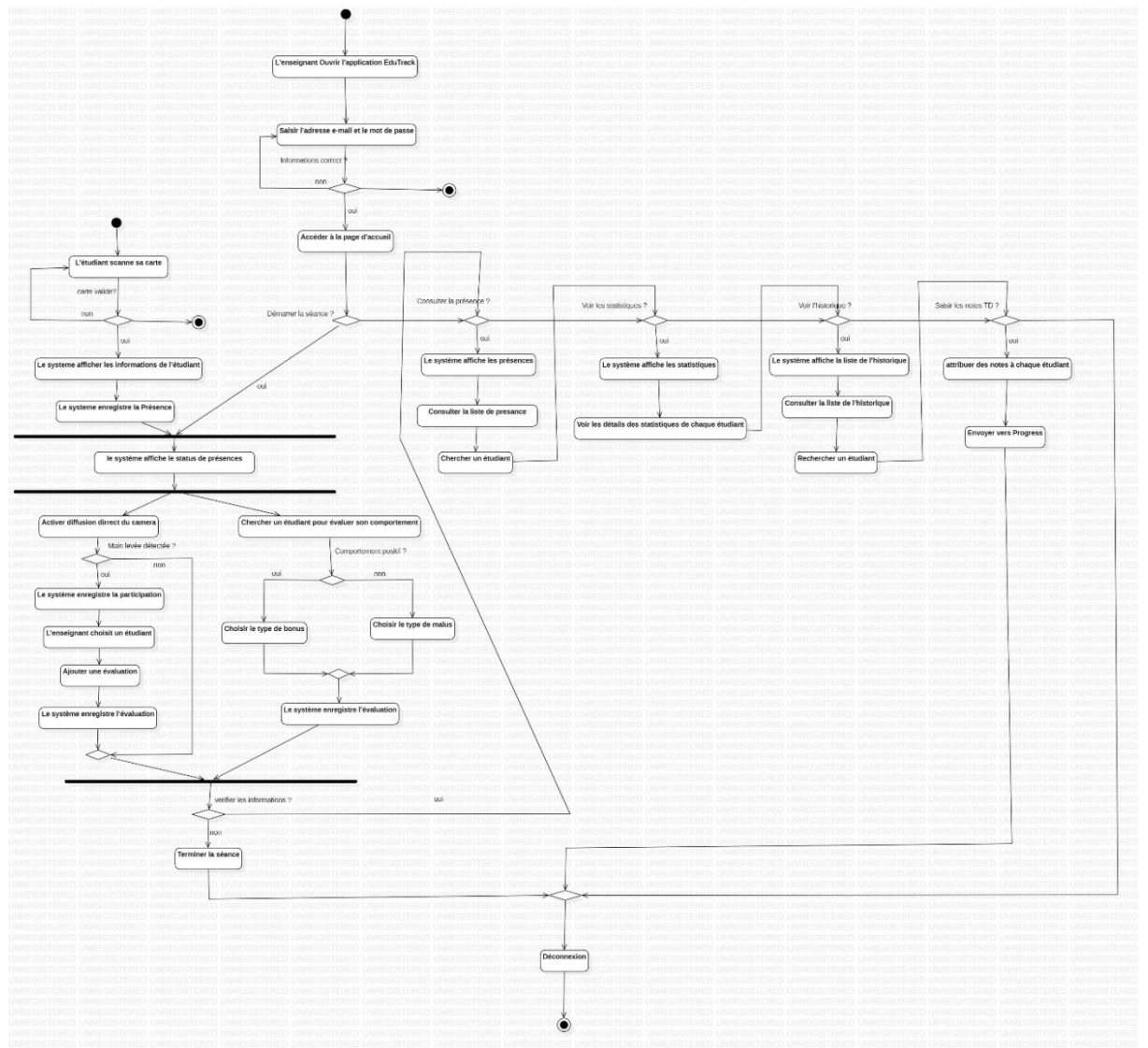


Figure 3.7 : Diagramme d'activités du système.

## 7. Conclusion

La phase de conception et d'analyse est une étape fondamentale à la réalisation du projet puisqu'elle facilite l'implémentation ainsi que la réalisation du système. Le prochain chapitre détaille les étapes de l'implémentation de l'application web de notre étude.

## **Chapitre 4 : Implementation**

## 1. Introduction

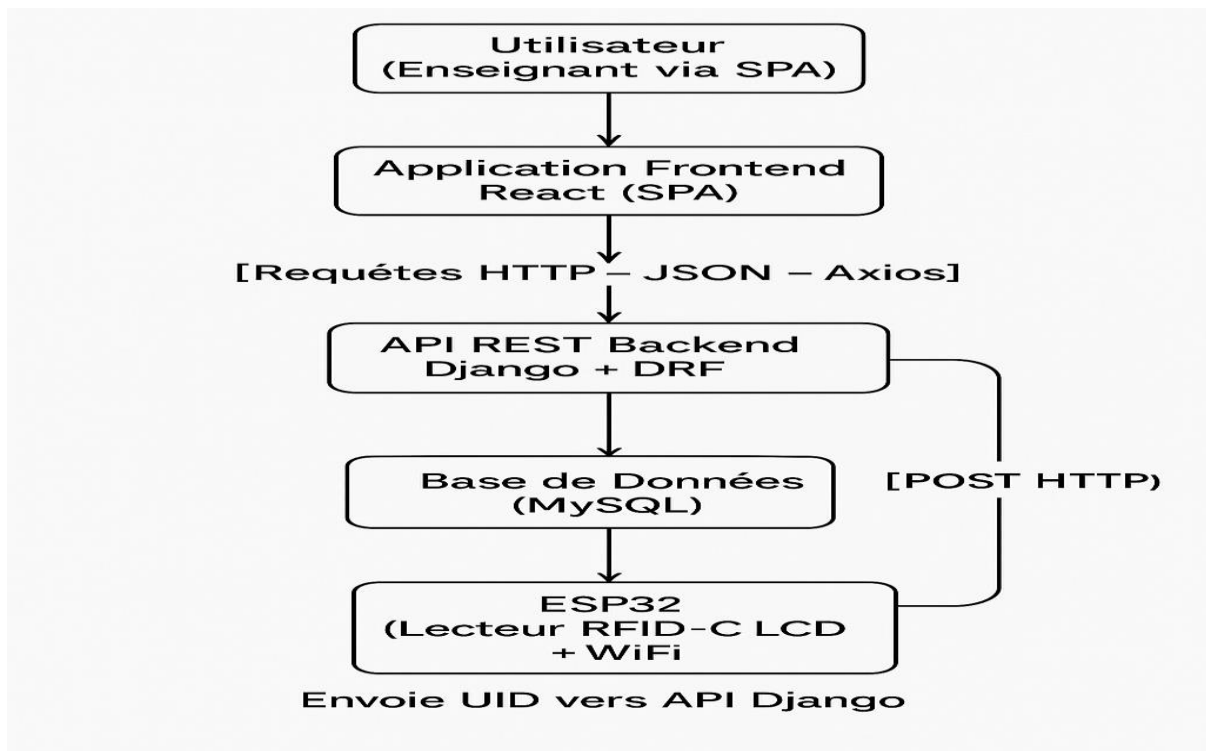
Le chapitre précédent a fait l'objet d'une description de l'étude conceptuelle du notre système proposé pour le suivi d'engagement des étudiants dans l'université. Le présent chapitre se concentre sur l'implémentation du système, nous présenterons les outils et technologies utilisés pour développer le système EduTrack, et nous concluons ce chapitre par une présentation des interfaces du système.

## 2. L'architecture du système

Notre projet EduTrack repose sur une architecture client-serveur moderne, où les rôles du client et du serveur sont bien distincts. Le client est une application web développée sous forme de SPA (Single-Page Application) à l'aide de React, ce qui permet une expérience utilisateur fluide, sans rechargement de page. Toutes les interactions avec le serveur se font via des requêtes HTTP, permettant au client d'envoyer ou de récupérer les données à la demande.

Le serveur, basé sur le Framework Django, agit en tant que backend central. Il expose une API RESTful qui traite les requêtes du client et répond avec des données structurées (généralement en format JSON). Cette architecture favorise la modularité, la scalabilité, ainsi que l'indépendance entre l'interface utilisateur et la logique métier, tout en offrant une base robuste pour intégrer des dispositifs IoT. C'est notamment le cas du module ESP32, qui joue le rôle d'un client physique connecté : il lit les cartes RFID des étudiants, envoie l'identifiant unique (UID) au serveur via le WiFi, et affiche la réponse (comme le nom de l'étudiant) sur un écran LCD. Cette intégration renforce l'automatisation du système de pointage et illustre la compatibilité du backend avec des capteurs connectés en temps réel.

**Figure 4.1** : Architecture générale du système EduTrack.



**Figure 4.1** : Architecture générale du système EduTrack.

## 3. Environnement du travail

### 3.1. Langage de développement

#### 3.1.1. Javascript

JavaScript (figure 4.1) est un langage de script léger, orienté objet, principalement utilisé pour rendre les pages web interactives. Il est exécuté côté client dans les navigateurs web, mais aussi dans d'autres environnements comme Node.js. JavaScript permet de manipuler le contenu d'une page web via le Document Object Model (DOM), de gérer des événements utilisateur (clics, survols, etc.) et de créer des animations, des transitions ou des applications dynamiques. [33] [34]



**Figure 4.2 :** JavaScript logo.

### 3.1.2. Python

Python (figure 4.1) est un langage de programmation informatique généraliste, interprété, à typage dynamique fort, qui favorise plusieurs paradigmes de programmation, notamment impérative structurée, fonctionnelle et orientée objet. Il est conçu pour être facile à apprendre et à utiliser, avec une syntaxe claire proche de l'anglais, utilisant l'indentation pour structurer le code au lieu d'accolades. [35]

Il est largement utilisé pour le développement web back-end, la création de logiciels, le scripting et l'automatisation, ainsi que pour la Data Science et le Machine Learning, domaines où il est particulièrement dominant grâce à ses nombreuses bibliothèques spécialisées.



**Figure 4.3 :** Python logo.

### 3.1.3. C/C++ (Arduino)

Le développement Arduino utilise principalement le langage C/C++ (figure 4.3) adapté aux microcontrôleurs.

- Base C : Le langage C est un langage procédural, simple et proche du matériel, ce qui permet de gérer précisément la mémoire et les ressources limitées des microcontrôleurs Arduino. Le code C est organisé en fonctions et manipule souvent directement la mémoire, ce qui exige une grande rigueur du développeur. [36]

- Extension C++ : Arduino utilise en réalité un compilateur C++ compatible avec la plupart du code C. Le C++ est un langage multiparadigme, combinant programmation procédurale, orientée objet et fonctionnelle. Cela permet aux développeurs Arduino d'adopter la POO quand cela est utile, tout en gardant la simplicité du C pour les parties proches du matériel. [37]

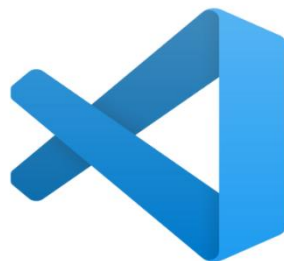


**Figure 4.4 :** C/C++ logo.

## 3.2. Logiciels utilisés

### 3.2.1. Visual Code Studio

Visual Studio Code (VS Code) (Figure 4.4) est un éditeur de code source développé par Microsoft, disponible sur Windows, Linux, MacOS et même en version web. Il est reconnu pour sa légèreté, sa rapidité et sa grande extensibilité grâce à un vaste écosystème d'extensions. Son interface bien pensée et son terminal intégré en font un outil très populaire auprès des développeurs, qu'ils soient professionnels ou amateurs. Il supporte nativement de nombreux langages comme JavaScript, Python, C++, Java, et peut en supporter beaucoup d'autres via des extensions. [38]



**Figure 4.5 :** VS Code logo.

La figure ci-dessous représente l'interface de Visual Code Studio :

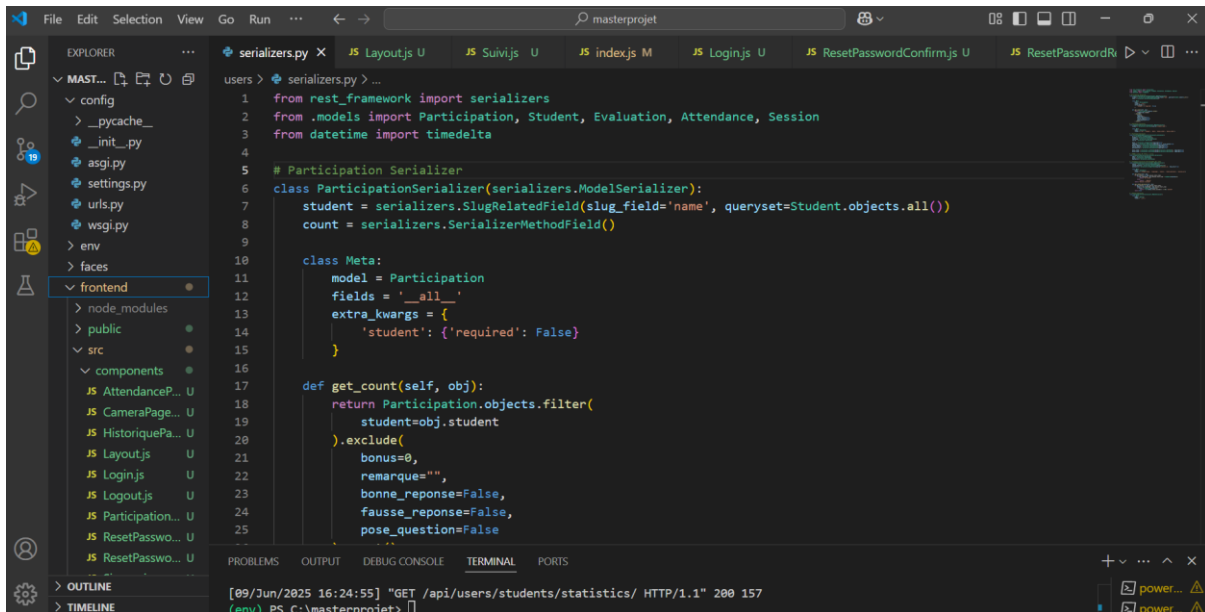


Figure 4.6: Interface Visual Studio Code.

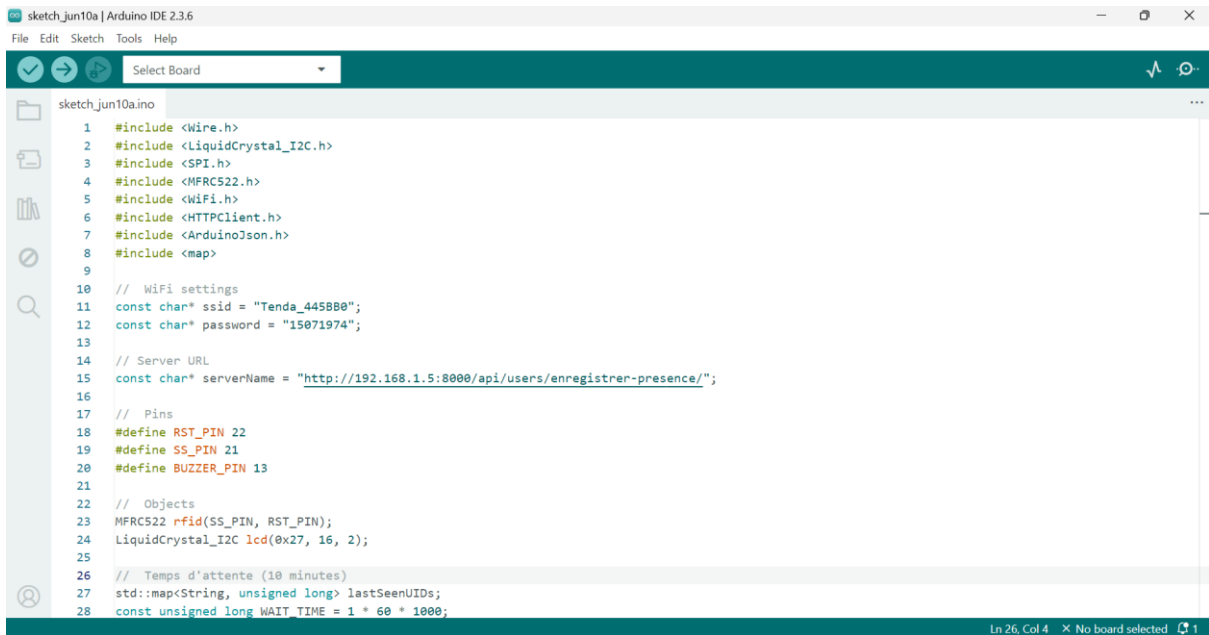
### 3.2.2. Arduino IDE

L'Arduino IDE (Figure 4.6) est un environnement de développement open source simple et efficace, spécialement conçu pour programmer facilement des cartes Arduino, ESP32 et autres microcontrôleurs compatibles. Il est disponible gratuitement sur Windows, MacOS et Linux. Le programme Arduino, appelé « sketch », est écrit dans l'éditeur, compilé en langage machine compréhensible par le microcontrôleur, puis téléversé sur la carte via USB. Pendant l'exécution, la carte peut communiquer avec l'ordinateur via le moniteur série, permettant d'envoyer ou recevoir des données. [39]



Figure 4.7 : Arduino IDE logo.

La figure ci-dessous représente l'interface de Arduino IDE :



```
sketch_jun10a.ino
1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3 #include <SPI.h>
4 #include <MFRC522.h>
5 #include <WiFi.h>
6 #include <HTTPClient.h>
7 #include <ArduinoJson.h>
8 #include <map>
9
10 // Wifi settings
11 const char* ssid = "Tenda_4458B0";
12 const char* password = "15071974";
13
14 // Server URL
15 const char* serverName = "http://192.168.1.5:8000/api/users/enregistrer-presence/";
16
17 // Pins
18 #define RST_PIN 22
19 #define SS_PIN 21
20 #define BUZZER_PIN 13
21
22 // Objects
23 MFRC522 rfid(SS_PIN, RST_PIN);
24 LiquidCrystal_I2C lcd(0x27, 16, 2);
25
26 // Temps d'attente (10 minutes)
27 std::map<String, unsigned long> lastSeenUIDs;
28 const unsigned long WAIT_TIME = 1 * 60 * 1000;
```

Figure 4.8 : Interface Arduino IDE.

### 3.3. Système de Gestion de Base de Données

#### 3.3.1. MySQL

MySQL (Figure 4.8) est un système de gestion de bases de données relationnelles (SGBDR) open source qui utilise le langage SQL pour créer, gérer et interroger des bases de données. Il stocke les données dans des tables organisées en lignes et colonnes, regroupées en schémas définissant la structure et les relations entre les tables. MySQL suit un modèle client-serveur : le serveur MySQL héberge les bases de données et traite les requêtes SQL envoyées par les clients, qui peuvent être des applications, interfaces graphiques ou scripts. Le serveur répond avec les résultats demandés, permettant ainsi la gestion centralisée des données. [40]



Figure 4.9 : MySQL logo.

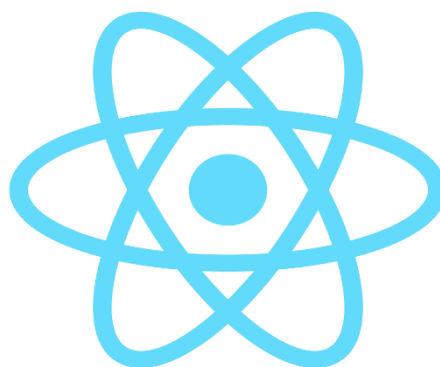
## 3.4. Framework utilisés

### 3.4.1. Django

Django est un Framework web open source de haut niveau écrit en Python, conçu pour simplifier et accélérer le développement d'applications web robustes et sécurisées. Il suit une architecture Modèle-Vue-Template qui sépare clairement la gestion des données, la logique métier et la présentation, facilitant ainsi la maintenance et l'évolution du code. Grâce à son ORM intégré, Django permet de manipuler les bases de données de manière intuitive sans avoir à écrire de requêtes SQL complexes. Ce Framework intègre également un système d'administration automatique, ce qui permet aux développeurs de gérer facilement le contenu de leurs applications sans effort supplémentaire. Connu pour ses nombreuses protections contre les vulnérabilités courantes du web, Django assure un haut niveau de sécurité dès la conception. [41] [42]

### 3.4.2. React

React est une bibliothèque JavaScript développée par Meta (anciennement Facebook) qui permet de créer des interfaces utilisateur interactives et performantes, principalement pour des applications web à page unique. Elle fonctionne en décomposant l'interface en composants réutilisables, chacun gérant son propre état et ses propriétés, ce qui facilite la maintenance et la modularité du code. React utilise une syntaxe appelée JSX, qui combine du JavaScript et du HTML, permettant d'écrire des composants de manière expressive et intuitive. React se concentre uniquement sur la couche de vue. Grâce à cette approche déclarative et à sa gestion efficace des mises à jour, React est largement adopté pour construire des applications web modernes, rapides et évolutives. [43] [44]



**Figure 4.10** : React logo.

## 3.5. Bibliothèques utilisées

### 3.5.1. Côté Backend (Django)

- Django REST Framework (DRF) : est une bibliothèque open-source puissante et flexible qui facilite la création d'API Web robustes avec Django. Elle permet de convertir facilement les données des modèles Django en formats JSON ou autres, de gérer les requêtes HTTP (GET, POST, PUT, DELETE) et d'assurer la sécurité des API grâce à des systèmes d'authentification intégrés. DRF propose également des outils pratiques comme les serializers pour la validation des données. Grâce à ces fonctionnalités, il est devenu un outil incontournable pour les développeurs souhaitant construire des applications web modernes et performantes basées sur des API RESTful. [45]

### 3.5.2. Côté Frontend (React.js)

- React Router DOM : est une bibliothèque essentielle pour gérer la navigation dans les applications React, permettant de créer des applications à page unique (SPA) avec un routage côté client fluide et performant. Elle fournit des composants comme BrowserRouter, Routes, et Route pour définir les chemins URL et associer des composants React à ces chemins. Cette bibliothèque est largement utilisée pour construire des interfaces utilisateur dynamiques et permet une navigation sans rechargement complet de la page, améliorant ainsi l'expérience utilisateur dans les applications React. [46]
- Ant Design (antd) : est une bibliothèque d'interface utilisateur (UI) basée sur React, conçue pour créer des applications web avec une expérience utilisateur riche et cohérente. Elle propose plus de 50 composants préfabriqués de haute qualité, tels que des boutons, des icônes, des mises en page, des formulaires, des tableaux et des éléments de navigation, permettant aux développeurs de construire rapidement des interfaces modernes et professionnelles. Cette bibliothèque est reconnue pour sa robustesse, sa documentation complète et sa communauté active. [47]
- Recharts : est une bibliothèque de graphiques pour React, construite avec React et D3, qui permet de créer facilement des visualisations de données interactives et responsives. Elle utilise une syntaxe déclarative basée sur des composants React, ce qui facilite la composition et la personnalisation des graphiques comme les courbes (LineChart), les barres (BarChart), les secteurs (PieChart), et bien d'autres. Recharts supporte aussi des fonctionnalités interactives comme les tooltips. Cette bibliothèque est légère, performante, et largement utilisée pour intégrer des graphiques dans des applications React modernes. [48]

### 3.5.3. Côté Caméra (Détection avec Python)

- La bibliothèque `face_recognition` : est un package Python très utilisé pour la détection et la reconnaissance faciale. Elle permet d'identifier les visages humains dans une image, d'extraire leurs encodages faciaux en vecteurs, et de comparer ces encodages pour reconnaître ou vérifier des individus. Elle utilise des modèles CNN pour une détection précise et peut fonctionner plus rapidement avec un GPU. Les fonctions principales incluent la détection des visages, la localisation des points clés du visage (landmarks), et le calcul des encodages faciaux, avec des options pour ajuster la précision et la vitesse. Cette bibliothèque est adaptée pour des applications de sécurité, d'authentification ou d'analyse d'images.
- MediaPipe : elle propose une solution spécifique pour la reconnaissance des gestes de la main en temps réel appelée `Gesture Recognizer`. Cette tâche permet de détecter et d'identifier différents gestes effectués avec la main à partir d'un flux vidéo, tout en fournissant les points de repère (landmarks) de la main détectée. Elle est disponible notamment via une API Python, avec un modèle pré-entraîné à télécharger et à intégrer facilement dans les projets. Cette solution est idéale pour des applications d'interaction homme-machine. MediaPipe peut détecter simultanément plusieurs mains et extraire 21 points clés par main, ce qui permet une analyse fine des mouvements et des gestes. [49] [50]
- OpenCV : est une bibliothèque open-source incontournable pour le traitement d'images et la vision par ordinateur. Elle permet de capturer, analyser et manipuler des images et des vidéos en temps réel, offrant une large gamme de fonctionnalités comme la détection d'objets, la reconnaissance faciale, le suivi de mouvements. Grâce à son interface Python simple et performante, OpenCV est largement utilisée dans des domaines variés tels que la robotique, la sécurité, la réalité augmentée ou les systèmes embarqués. Son efficacité et sa flexibilité en font un outil de choix pour les développeurs souhaitant intégrer des capacités de vision dans leurs applications.

### 3.5.4. Côté ESP32

- La bibliothèque `HTTPClient.h` : est une librairie Arduino qui facilite l'envoi de requêtes HTTP (GET, POST, PUT, DELETE) vers un serveur web, comme un serveur Django. Elle fonctionne avec différents types de clients réseau (`WiFiClient`) et gère automatiquement les en-têtes HTTP, le code de statut de la réponse, ainsi que la récupération du contenu. Par exemple, sur une carte ESP32 connectée en Wi-Fi, on peut utiliser `HTTPClient` pour envoyer une requête POST avec un payload JSON vers une API Django, récupérer la réponse et la traiter facilement. [51]

- La bibliothèque WiFi.h : est une librairie Arduino qui permet à une carte compatible (comme l'ESP32 ou l'ESP8266) de se connecter à un réseau Wi-Fi. Elle offre des fonctions pour gérer la connexion à un point d'accès, récupérer l'adresse IP, scanner les réseaux disponibles, et gérer les événements liés au Wi-Fi (connexion, déconnexion, obtention d'adresse IP). La bibliothèque est automatiquement incluse avec l'extension ESP32 dans l'IDE Arduino, et est essentielle pour développer des projets IoT connectés à Internet. Elle supporte aussi la création de serveurs et clients TCP via des classes comme WiFiServer et WiFiClient. Cette bibliothèque est documentée officiellement par Arduino et largement utilisée pour programmer des modules Wi-Fi dans des applications embarquées.

## 3.6. Matériels utilisés

### 3.6.1. ESP32

L'ESP32 est un microcontrôleur très populaire intégrant une connectivité Wi-Fi et Bluetooth, largement utilisé pour des projets IoT. Avec la bibliothèque Arduino WiFi.h, il est possible de scanner les réseaux Wi-Fi à proximité, de se connecter à un réseau existant, et de gérer la connexion de manière fiable.

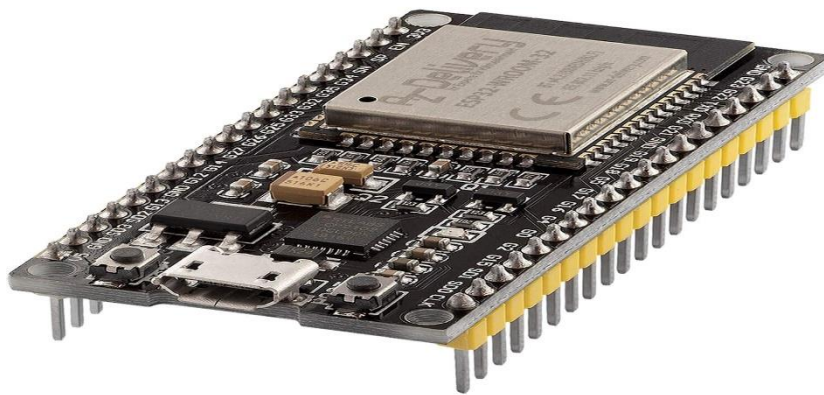


Figure 4.11 : ESP32.

### 3.6.2. RFID RC522

Le module RFID RC522 est un lecteur très populaire, souvent utilisé avec des microcontrôleurs comme l'ESP32 pour des projets d'identification et de contrôle d'accès. Fonctionnant principalement via le protocole SPI, il permet de lire l'identifiant unique (UID) des cartes RFID compatibles à 13,56 MHz, ainsi que d'écrire et de gérer des données sur ces cartes. Grâce à sa simplicité d'intégration et à la bibliothèque MFRC522 disponible dans l'environnement Arduino, il est facile de développer des applications telles que l'authentification sans fil. [52]



**Figure 4.12 :** RFID RC522.

### 3.6.3. Écran LCD

Un écran LCD 16x2 est un afficheur à cristaux liquides capable d'afficher 16 caractères sur 2 lignes. Il est souvent utilisé avec des microcontrôleurs pour afficher des messages, des données ou des valeurs de capteurs. La version avec interface I2C simplifie grandement le câblage en utilisant seulement quatre connexions : alimentation (VCC, GND) et communication (SDA, SCL). Ce type d'écran nécessite l'installation de la bibliothèque Arduino LiquidCrystal\_I2C pour faciliter la programmation. [53]



**Figure 4.13 :** Écran LCD 16x2.

## 4. Interfaces de l'application

Dans cette partie, nous présentons quelques interfaces de l'application EduTrack :

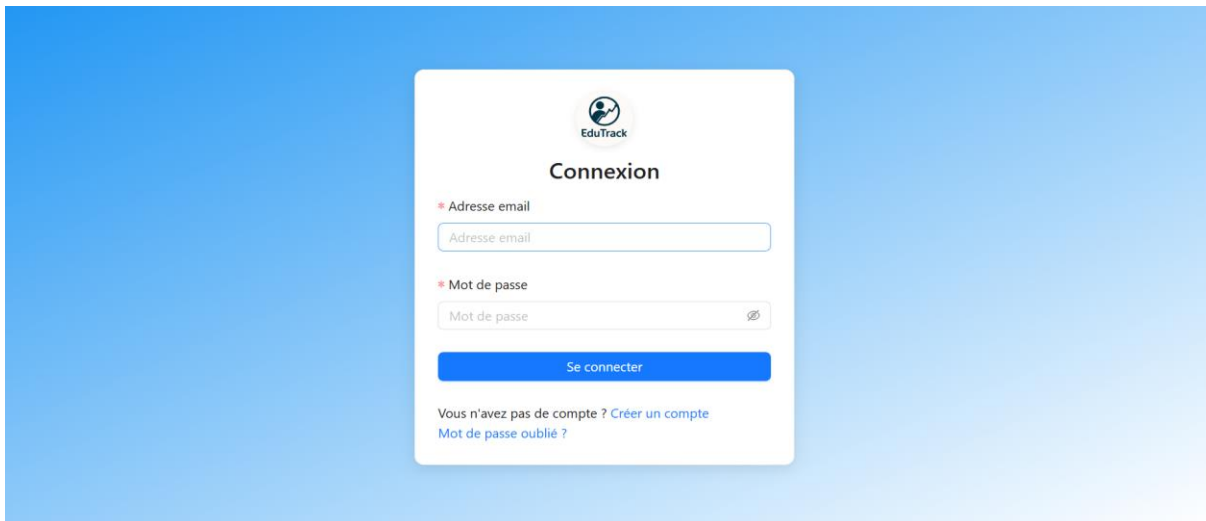


Figure 4.14 : Interface de connexion.

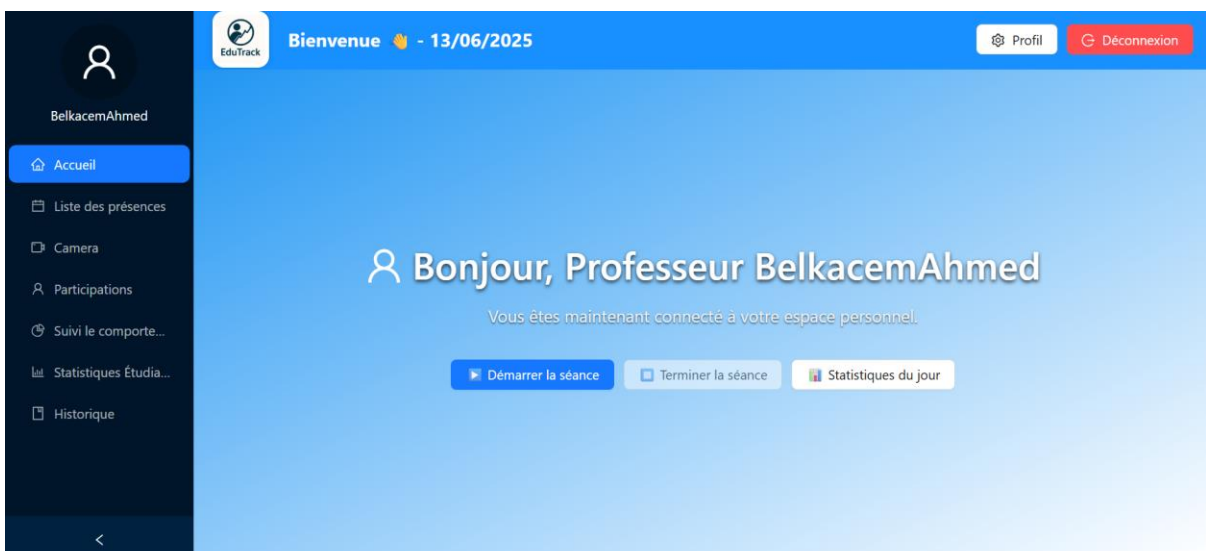


Figure 4.15 : Page d'accueil du système EduTrack.

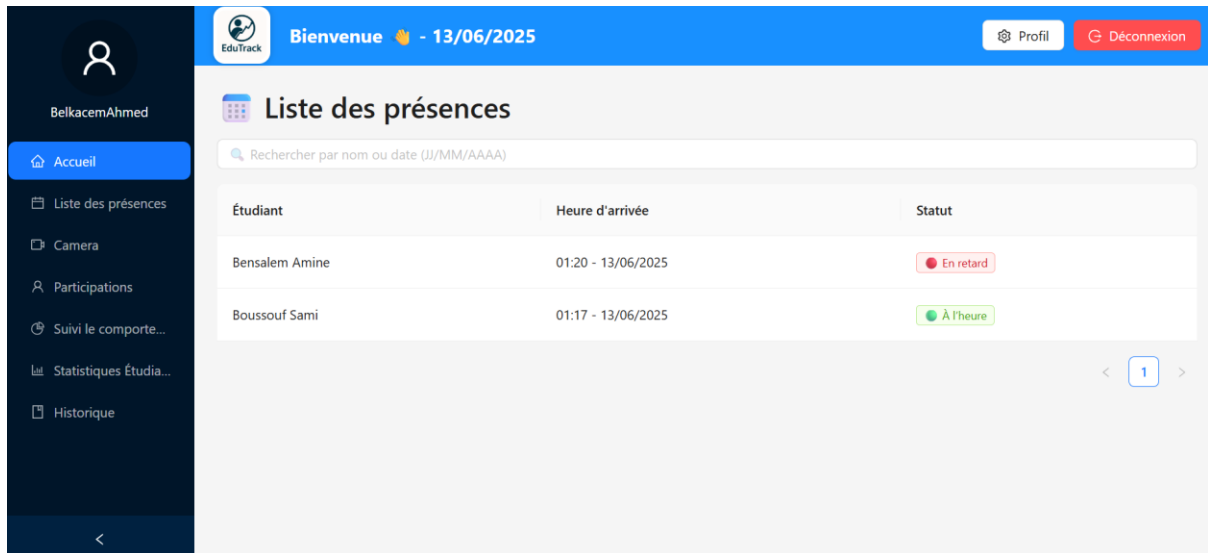


Figure 4.16 : Interface liste des présences.

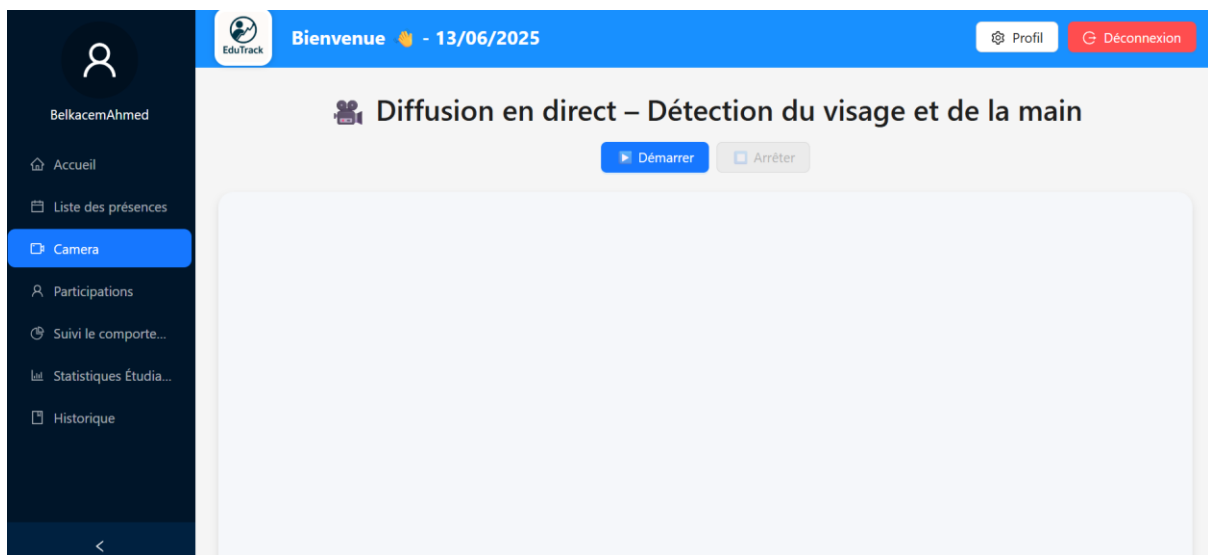


Figure 4.17 : Interface de diffusion en direct du vidéo.

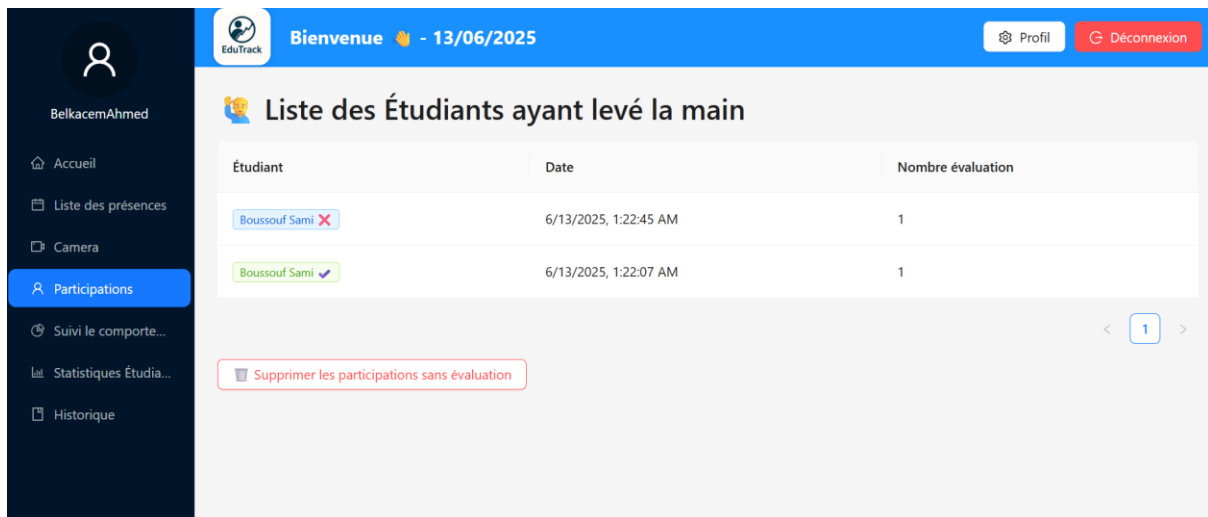


Figure 4.18 : Interface liste des participations.

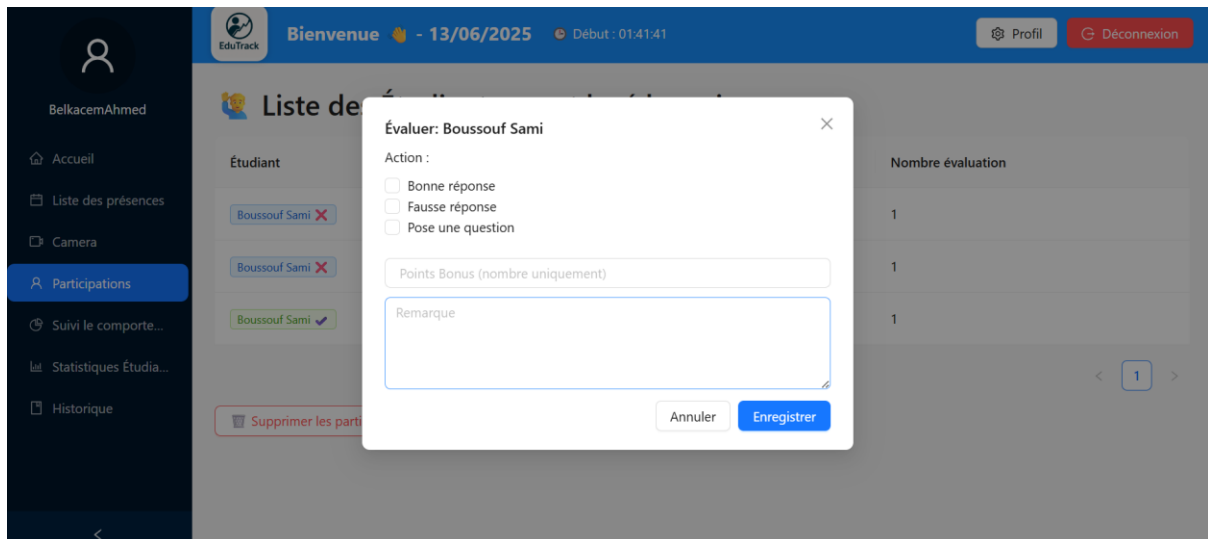


Figure 4.19 : Interface d'évaluation de participation.

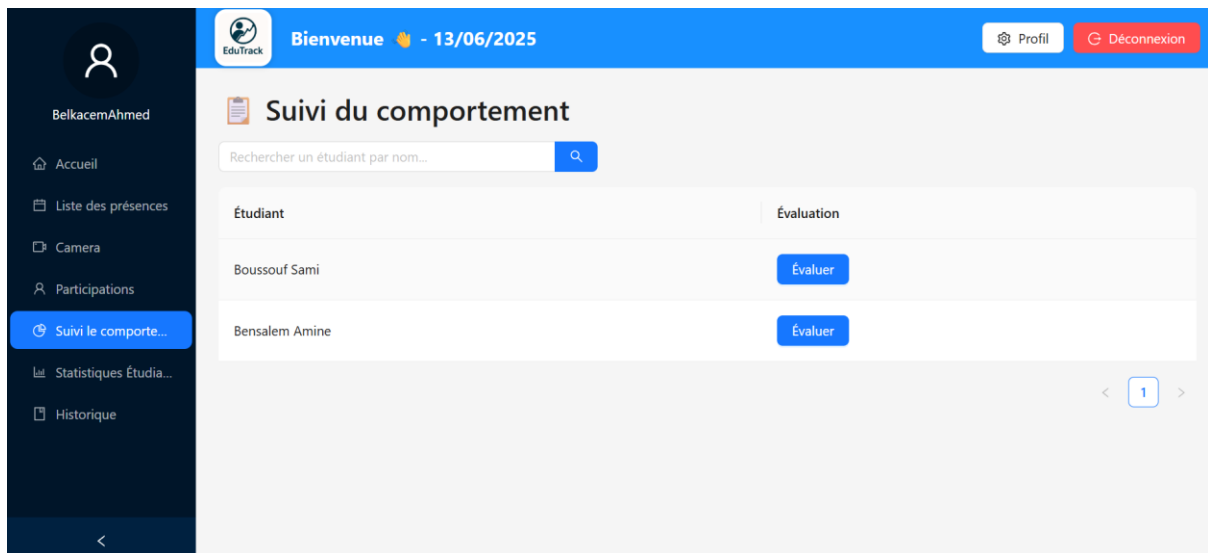


Figure 4.20 : Interface de suivi de comportement.

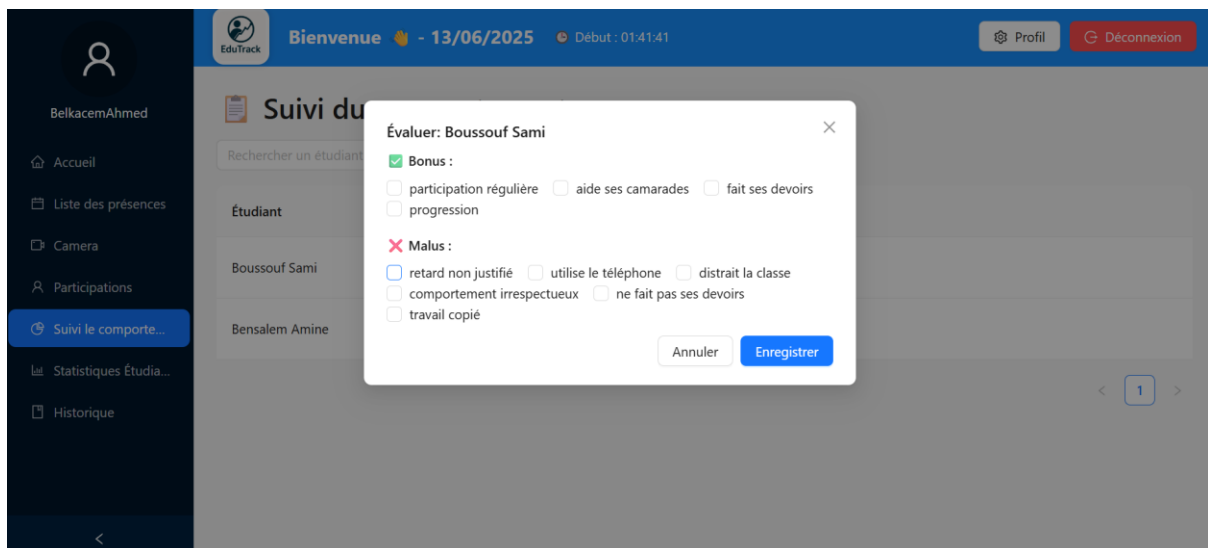


Figure 4.21 : Interface d'évaluation du comportement.

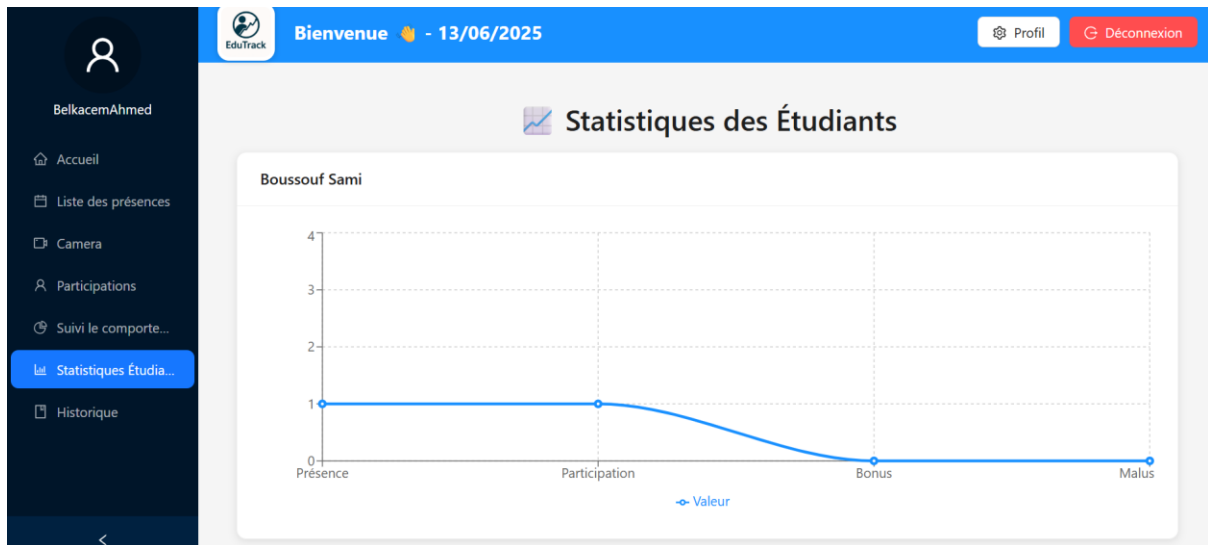


Figure 4.22 : Interface de statistique des étudiants.

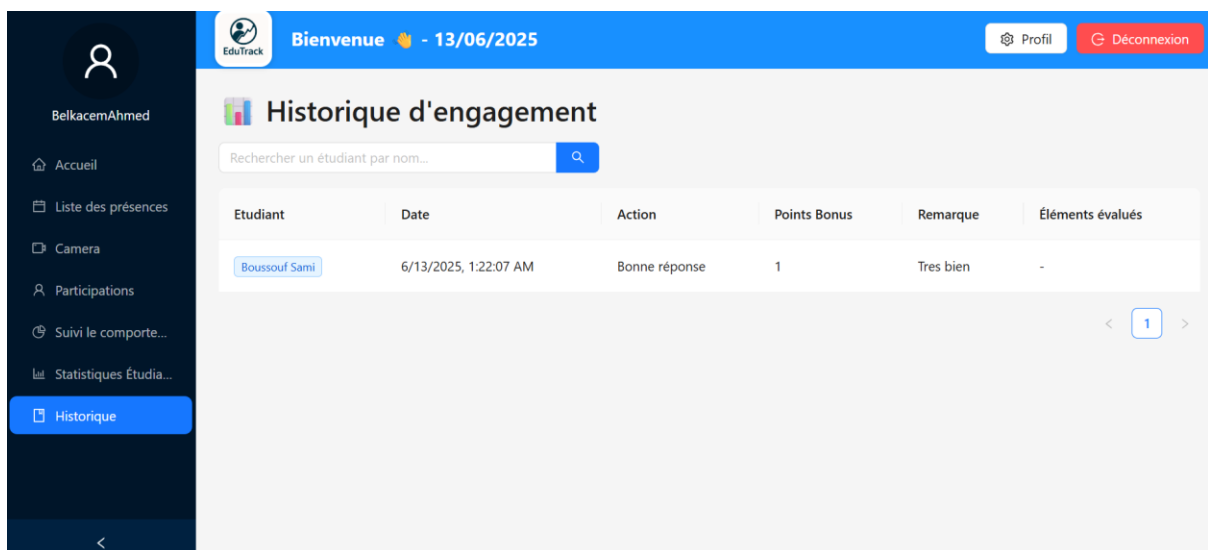


Figure 4.23 : Interface d'historique.

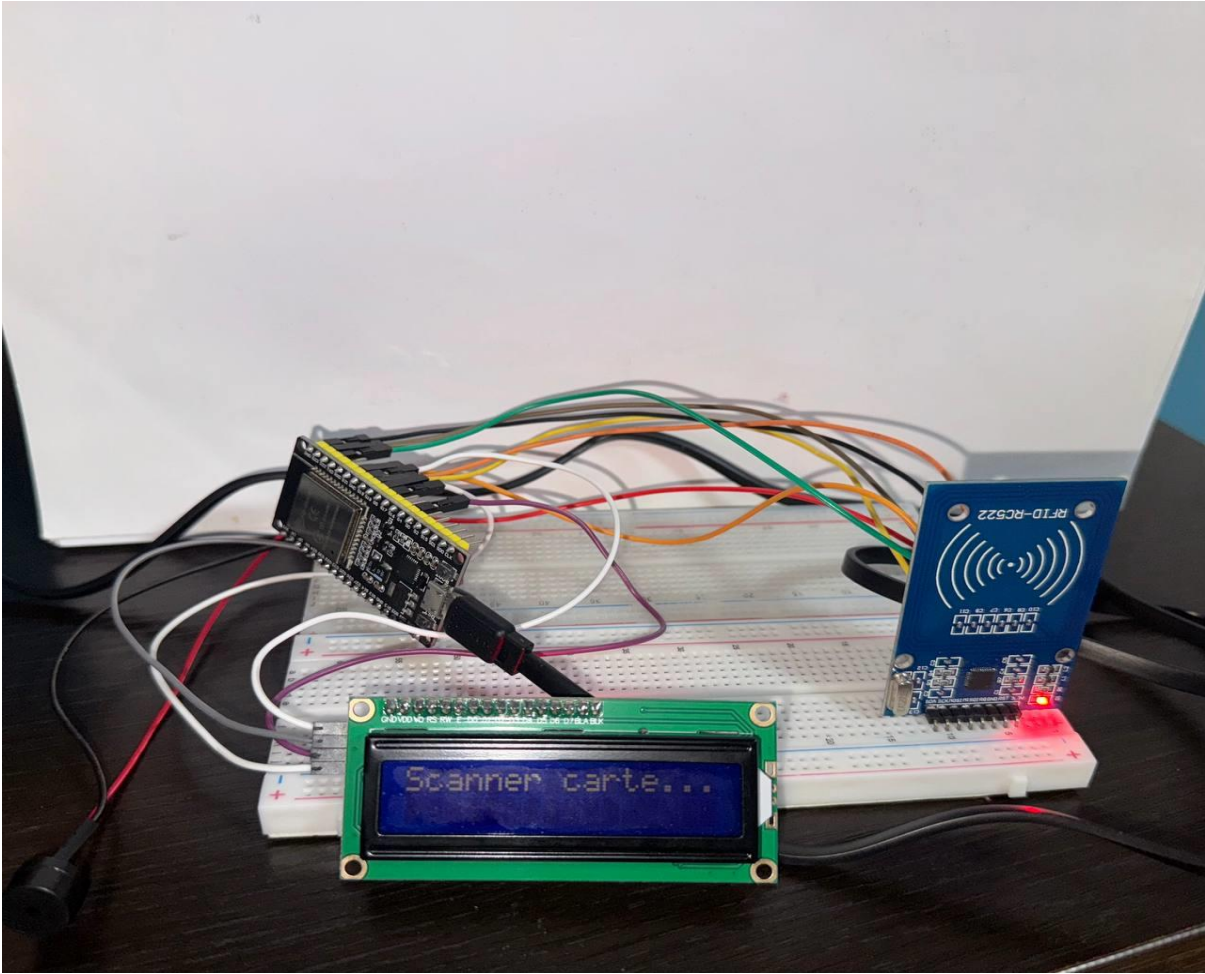


Figure 4.24 : Le Système avant scanner la carte.

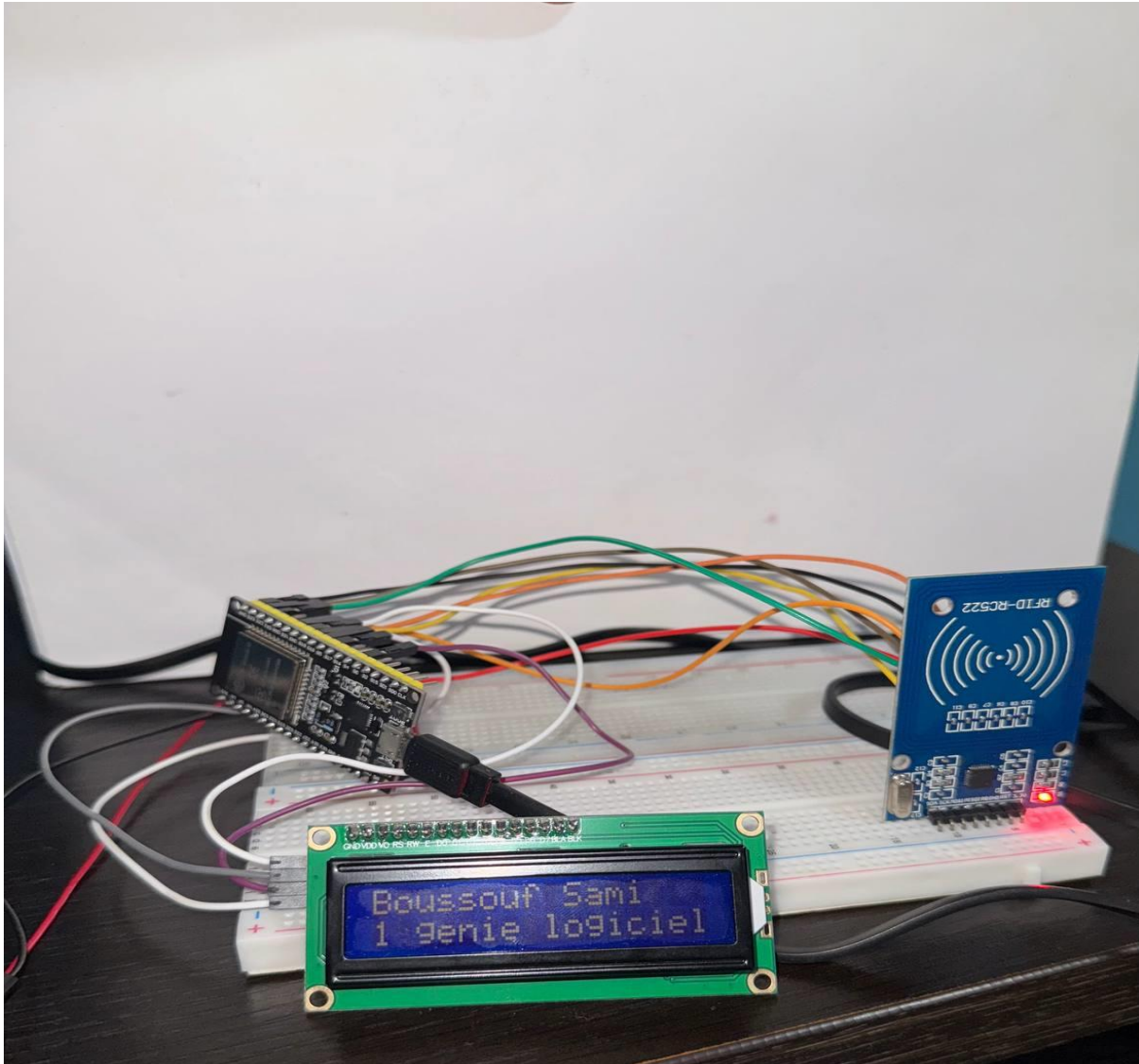


Figure 4.25 : Le Système Après scanner la carte.

## 5.Conclusion

Dans ce chapitre, nous avons présenté l'architecture et les outils et langages nécessaires au développement du système EduTrack, et nous avons finis par la présentation de quelques interfaces de l'application.

## **Conclusion Générale**

### Conclusion générale

Ce mémoire a présenté le développement d'un système intelligent, **EduTrack**, visant à automatiser la gestion des absences et à assurer un suivi objectif de l'engagement des étudiants dans le cadre universitaire. Face aux limites des méthodes traditionnelles – souvent chronophages, peu fiables et subjectives – le recours à l'Internet des Objets (IoT) s'est imposé comme une solution innovante et efficace.

Grâce à l'intégration de technologies telles que les cartes **RFID** pour la présence et les caméras pour l'observation de la participation active, EduTrack permet une collecte automatisée de données en temps réel, offrant aux enseignants une vision plus claire et équitable du comportement des étudiants. Ce système contribue ainsi à une amélioration notable de la qualité pédagogique, en facilitant la prise de décision et en renforçant l'interaction entre les apprenants et les encadrants.

En somme, **EduTrack** illustre le potentiel des technologies connectées dans la transformation de l'enseignement supérieur vers un modèle plus intelligent, transparent et centré sur l'étudiant.

### Bibliographie

- [1] Baland, C., Cauquil, D., Gayet, T., Juvigny, J., Lifchitz, R., & Nguyen, N. (s.d.). La sécurité de l'Internet des Objets [Livre blanc].
- [2] Cluster of European Research Projects on the Internet of Things. (2010, mars). Vision and challenges for realising the Internet of Things.
- [3] Haddab, Y. (s.d.). Introduction à l'internet des objets (IdO – IoT). Université de Montpellier.
- [4] Orange. (s.d.). Objet connecté. <https://iotjourney.orange.com/fr-FR/explorer/les-solutions-iot/objet-connecte>
- [5] DDruid. (s.d.). Produit intelligent. <https://ddruid.io/produit-intelligent/>
- [6] Siemens. (s.d.). Histoire de l'IoT. <https://www.siemens.com/fr/fr/entreprise/stories/innovation-technologies/jumeau-numerique/histoire-iot.html>
- [7] Synox. (s.d.). Internet des objets (IoT), qu'est-ce que c'est ?. <https://www.synox.io/actualites-sectorielle/internet-des-objets-iot-qu-est-ce-que-cest/>
- [8] DDruid. (s.d.). IoT. <https://ddruid.io/iot/>
- [9] Evans, D. (2011, avril). L'Internet des objets : comment l'évolution actuelle d'Internet transforme-t-elle le monde ? Cisco Internet Business Solutions Group (IBSG).
- [10] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660.
- [11] Académie d'Amiens. (s.d.). Capteurs et actionneurs. [http://sti.ac-amiens.fr/IMG/pdf/s2\\_capteurs\\_et\\_actionneurs-2.pdf](http://sti.ac-amiens.fr/IMG/pdf/s2_capteurs_et_actionneurs-2.pdf)
- [12] Ray, P. P. (2018). A survey on Internet of Things architectures. *Journal of King Saud University – Computer and Information Sciences*, 30(3), 291–319.
- [13] WitBlox. (s.d.). Arduino UNO R3 Board - Clone Model - High Quality (with cable). <https://witblox.com/products/arduino-uno-r3-development-board-clone-model-high-quality-with-cable>
- [14] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347–2376.
- [15] Banks, A., & Gupta, R. (2014). MQTT Version 3.1.1. OASIS Standard. <https://docs.oasis-open.org/mqtt>
- [16] Shelby, Z., Hartke, K., & Bormann, C. (2014). The Constrained Application Protocol (CoAP) (RFC 7252). IETF. <https://datatracker.ietf.org/doc/html/rfc7252>

## Bibliographie

---

- [17] Saqib, S. N. U. (2023, novembre 8). IoT Communication Protocols. <https://www.linkedin.com/pulse/iot-communication-protocols-syed-najam-us-saqib-elhyf>
- [18] Minerva, R., Biru, A., & Rotondi, D. (2015). Towards a definition of the Internet of Things (IoT). IEEE Internet Initiative.
- [19] Skandrani, F., & Skandrani, F. (2023, octobre 10). Architecture IoT : L'essentiel à savoir. IoT Industriel. <https://iotindustriel.com/iot-iiot/architecture-iot-essentiel-a-savoir/>
- [20] Mittal, P., Sharma, M., & Thulal, A. N. (2023). IoT Challenges, Solution Framework, and Implementation. EAI Endorsed Transactions on Internet of Things, 9(3), e4. <https://doi.org/10.4108/eetiot.v9i3.3408>
- Références (suite)**
- [21] Idéematic. (2023, février 14). Les avantages de l'utilisation de l'Internet des Objets (IoT) dans les applications mobiles. <https://www.ideematic.com/actualites/2023/02/les-avantages-de-lutilisation-de-linternet-des-objets-iot-dans-les-applications-mobiles/>
- [22] Fortinet. (s.d.). Qu'est-ce que l'IoT ? Définition, avantages et défis. <https://www.fortinet.com/fr/resources/cyberglossary/iot>
- [23] Badshah, S., Muhammad, N., Ullah, A., & Ahmad, M. (2023). Towards Smart Education through the Internet of Things: A Review. arXiv preprint arXiv:2304.01337.
- [24] Fernández-Batanero, J. M., Román-Graván, P., Reyes-Rebollo, M. M., & Montenegro-Rueda, M. (2022). Adoption of the Internet of Things in higher education: The role of security, privacy and support for learning. *Education and Information Technologies*, 27(3), 3403–3422.
- [25] Gilman, E., Isomursu, M., et al. (2020). Internet of Things for Smart Spaces: A University Campus Case Study. *Journal of Ambient Intelligence and Humanized Computing*.
- [26] Mircea, M., Stoica, M., & Ghilic-Micu, B. (2021). Investigating the impact of the Internet of Things in higher education environment. *IEEE Access*, 9, 33396–33409.
- [27] Ibrahim, I. S., & Kenwright, B. (2022). Smart education: Higher education instruction and the Internet of Things (IoT). arXiv preprint arXiv:2207.02585.
- [28] Kim, H., Park, Y., & Lee, J. (2019). RFID-based attendance management system for university students. *IEEE Access*, 7, 18735–18747.
- [29] Romero-Rodríguez, J.-M., Aznar-Díaz, I., Hinojo-Lucena, F.-J., & Cáceres-Reche, M.-P. (2020). Smart campuses and security: Towards an intelligent university environment. *Sensors*, 20(3), 779.
- [30] Xie, X., & Yang, C. (2021). IoT-based model for intelligent innovation practice system in higher education institutions. *Journal of Physics: Conference Series*, 1914(1), 012068. <https://doi.org/10.1088/1742-6596/1914/1/012068>

## Bibliographie

---

- [31] Dake, B., Bada, G., & Dadzie, D. (2022). Internet of Things (IoT) applications in education: Benefits and challenges in Ghana. *Journal of Information Technology Education: Research*, 21, 311–338.
- [32] Aiswarya, S., Aditya, K. S., & Jha, G. K. (2021). Teachers' perspective on the challenges and opportunities of e-learning implementation in higher education using IoT and digital tools. *Education and Information Technologies*, 26(6), 7639–7655.
- [33] MDN Web Docs. (n.d.). JavaScript | MDN. <https://developer.mozilla.org/fr/docs/Web/JavaScript>
- [34] Amazon Web Services, Inc. (n.d.). Qu'est-ce que JavaScript ? – JavaScript expliqué – AWS. <https://aws.amazon.com/fr/what-is/javascript/>
- [35] Amazon Web Services, Inc. (n.d.). Qu'est-ce que Python ? – Le langage Python expliqué – AWS. <https://aws.amazon.com/fr/what-is/python/>
- [36] Brissonneau, F. (2023, novembre 23). C, C++ et C#... Quelles différences entre ces langages de programmation ? ORSYS Le Mag. <https://orsys-lemag.com/c-c-et-c-quelles-differences-entre-ces-langages/>
- [37] Microsoft. (2021, novembre 3). Documentation for Visual Studio Code. <https://code.visualstudio.com/docs>
- [38] (n.d.). Logiciel : IDE – Arduino : l'essentiel. <https://arduino.blaise-pascal.fr/presentation/logiciel/>
- [39] D. G. (2025, avril 28). What is MySQL and how does it work. Hostinger Tutorials. <https://www.hostinger.com/tutorials/what-is-mysql>
- [40] MDN Web Docs. (n.d.). Introduction à Django - Apprendre le développement web. [https://developer.mozilla.org/fr/docs/Learn\\_web\\_development/Extensions/Server-side/Django/Introduction](https://developer.mozilla.org/fr/docs/Learn_web_development/Extensions/Server-side/Django/Introduction)
- [41] Kassel, R. (2023, novembre 9). Django : Tout sur le framework de développement web en Python. DataScientest. <https://datascientest.com/django-tout-savoir>
- [42] Links, I. (n.d.). Qu'est-ce que ReactJS et pourquoi devrions-nous utiliser ReactJS ? <https://ibracilinks.com/blog/quest-ce-que-reactjs-et-pourquoi-devrions-nous-utiliser-reactjs>
- [43] Chakir, C. (2025, avril 9). Qu'est-ce que React, comment l'utiliser et le déployer ? Hostinger Tutoriels. <https://www.hostinger.com/fr/tutoriels/quest-ce-que-react>
- [44] Le Codeur, E. (2021, septembre 17). Votre première introduction à Django REST Framework. DEV Community. <https://dev.to/ericcodeur/votre-premiere-introduction-a-django-rest-framework-j8k>
- [45] Olawanle, J. (2024, juillet 19). A complete guide to routing in React. Hygraph. <https://hygraph.com/blog/routing-in-react>

## Bibliographie

---

[46] Acharya, D. P. (2023, octobre 12). Bibliothèques de composants React UI : Nos meilleurs choix pour 2022. Kinsta®. <https://kinsta.com/fr/blog/bibliotheque-composants-react-ui/>

[47] Ant Design. (n.d.). Components Overview - Ant Design. <https://ant.design/components/overview>

[48] CodeParrot. (n.d.). ReCharts: The Ultimate React Charting Library. <https://codeparrot.ai/blogs/recharts-the-ultimate-react-charting-library>

[49] Google AI for Developers. (n.d.). Gesture recognition guide for Python. [https://ai.google.dev/edge/mediapipe/solutions/vision/gesture\\_recognizer/python](https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer/python)

[50] Talbi, I. (2025, juin 5). Mediapipe : le framework révolutionnaire en vision par ordinateur. La Revue IA. <https://larevueia.fr/mediapipe-le-framework-revolutionnaire-en-vision-par-ordinateur/>

[51] Santos, R., & Santos, R. (2022, octobre 27). ESP32 HTTP GET and HTTP POST with Arduino IDE. Random Nerd Tutorials. <https://randomnerdtutorials.com/esp32-http-get-post-arduino/>

[52] (n.d.). Lecture de l'identificateur de la carte RFID en utilisant ESP32 et RFID-RC522. <https://www.robotique.site/tutoriel/lecture-de-lidentificateur-de-la-carte-rfid-en-utilisant-esp32-et-rfid-rc522/>

[53] (n.d.). La carte Arduino UNO et l'afficheur LCD I2C 2X16. <https://www.robotique.site/tutoriel/la-carte-arduino-uno-et-lafficheur-lcd-i2c-2x16/>



