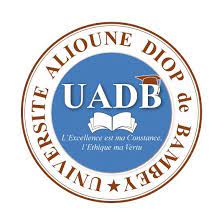
**REPUBLQUE DU SENEGAL**



**(Un Peuple – Un But – Une Foi)**

**M**inistère de l’**E**nseignement **S**upérieur, de la **R**echerche et de l’**I**nnovation **(MESRI)**

**U**niversité **A**lioune **D**iop De **B**ambey **(UADB)**



**UFR** : **S**ciences **A**ppliquées et **T**echnologies de l’**I**nformation et de la **C**ommunication **(SATIC)**

**Département** : **T**echnologies de l’**I**nformation et de la **C**ommunication **(TIC)**

**Licence Professionnel** : **S**ystème **R**éseaux et **T**élécoms **(SRT)**

THEME : CONCEPTION D’UN SYSTEME DE POINTAGE MULTIFONCTIONNEL

**Présenté par : Encadré par :**

Maguette Seye Mr Ouesse

Mamadou Absa Gueye

Année Académique : 2022-2023

REPLI1QUE de le page de garde

REMERCIEMENTS ET DEDICACES

SIGLES

**USB** : Universal Serial Bus

**IoT** : Internet of Things

**ESP** : Espressif Systems Product

**WEP** : Wired Equivalent Privacy

**WPA** : Wi-Fi Protected Access

**GPIO** : General Purpose Input/Output.

**VIN** : Voltage IN

**GND** : Ground

**RX** : Receive

**TX** : Transfer

**SPI** : Serial Peripheral Interface

**I2C** : Inter-Integrated Circuit

**RFID** : Radio Frequency Identification

**HTML** : Hypertext Markup Language

**CSS** : Cascading Style Sheets

**UML** : Unified Modeling Language

**ORM** : Object Relational Mapping

**IDE** : Integrated Development Environment

**UART** :

TABLE DE MATIERE

Table des matières

[Chapitre I : Présentation générale 11](#_Toc170234051)

[I.1 Présentation du sujet 11](#_Toc170234052)

[I.1.1 Contexte Générale 11](#_Toc170234053)

[I.1.2 Problématique 12](#_Toc170234054)

[I.1.3 Objectifs 12](#_Toc170234055)

[I.2 Présentation des technologies : 12](#_Toc170234056)

[I.2.1 Historiques de Arduino : 12](#_Toc170234057)

[I.2.2 Les autres cartes : 13](#_Toc170234058)

[I.3 Les outils utilisés : 15](#_Toc170234059)

[I.3.1 Esp8266 15](#_Toc170234060)

[I.4 RFID : 16](#_Toc170234061)

[I.4.1 Cartes RFID 16](#_Toc170234062)

[I.4.2 Tags RFID 17](#_Toc170234063)

[I.4.3 Lecteurs RFID 17](#_Toc170234064)

[I.4.4 Principe de fonctionnement 18](#_Toc170234065)

[I.5 L’empreinte digitale R307S 18](#_Toc170234066)

[I.5.1 Mécanisme de la capture d’empreinte 19](#_Toc170234067)

[I.6 La platine 19](#_Toc170234068)

[I.7 Le buzzer 20](#_Toc170234069)

[Chapitre II : Langages et outils de développement utilisés : 20](#_Toc170234070)

[II.1 IDE Arduino 20](#_Toc170234072)

[II.2 Visual Studio Code 21](#_Toc170234073)

[II.3 Python 22](#_Toc170234074)

[II.4 Django 22](#_Toc170234075)

[II.4.1 Définition 22](#_Toc170234076)

[II.4.2 Fonctionnement 23](#_Toc170234077)

[II.5 Django base donnée et ORM 25](#_Toc170234078)

[II.6 HTML 25](#_Toc170234079)

[II.7 CSS 26](#_Toc170234080)

[II.8 JavaScript 26](#_Toc170234081)

[II.9 Bootstrap 26](#_Toc170234082)

[II.10 Font Awesome 27](#_Toc170234083)

[II.11 UML 27](#_Toc170234084)

[Chapitre III : Conception détaillée du dispositif de pointage 28](#_Toc170234085)

[III.1 Architecture générale 28](#_Toc170234087)

[III.2 CONFIGURATION DE ESP8266 POUR ARDUINO : 28](#_Toc170234088)

[III.3 Branchement des composants 30](#_Toc170234089)

[III.3.1 Esp8266 et rfid 30](#_Toc170234090)

[III.3.2 Esp8266 et R307s 30](#_Toc170234091)

[III.3.3 Esp8266 et buzzer 30](#_Toc170234092)

[III.3.4 Branchement de tous les composants 31](#_Toc170234093)

[III.4 Présentation de la page web 31](#_Toc170234094)

[Figure I‑1 : Arduino 13](#_Toc170234095)

[Figure 2 : Kit Arduino 13](#_Toc170234096)

[Figure 3 : esp8266 13](#_Toc170234097)

[Figure 4 : image raspberry 14](#_Toc170234098)

[Figure 5 : L’illustration de l’esp8266 avec ses broches : 15](#_Toc170234099)

[Figure 6 :RFID 16](#_Toc170234100)

[Figure 7 : La carte rfid 17](#_Toc170234101)

[Figure 8 : tags rfid 17](#_Toc170234102)

[Figure 9 : lecteur rfid 18](#_Toc170234103)

[Figure 10 : principe de fonctionnement de rfid 18](#_Toc170234104)

[Figure 11 : R307s 19](#_Toc170234105)

[Figure 12 : Mécanisme de la capture d’empreinte 19](#_Toc170234106)

[Figure 13 : la platine 19](#_Toc170234107)

[Figure 14 : : Illustration Arduino IDE 21](#_Toc170234108)

[Figure 15 : logo Visual Studio Code 21](#_Toc170234109)

[Figure 16 :logo python 22](#_Toc170234110)

[Figure 17 :logo Django 23](#_Toc170234111)

[Figure 18 : fonctionnement de django 24](#_Toc170234112)

[Figure 19 : logo Font Awesome 25](#_Toc170234113)

[Figure 20 : illustration de ORM 25](#_Toc170234114)

[Figure 21 : logo HTML 25](#_Toc170234115)

[Figure 22 : logo css 26](#_Toc170234116)

[Figure 23 : logo JavaScript 26](#_Toc170234117)

[Figure 24 : logo Bootstrap 26](#_Toc170234118)

[Figure 25 : font awesome 27](#_Toc170234119)

[Figure 26 : illustration UML 28](#_Toc170234120)

[Figure 27 :branchement esp8266 et rfid 30](#_Toc170234121)

[Figure 28 : branchement esp8266 et r307s 30](#_Toc170234122)

[Figure 29 : Branchement esp8266 et le buzzer 31](#_Toc170234123)

[Tableau I-1 : Comparaison de l’Arduino, Esp et Raspberry Pi 14](#_Toc170175819)

INTRODUCTION :

L'IoT, ou Internet des Objets, est un concept technologique qui désigne la connectivité et l'interopérabilité des objets physiques à travers Internet. Cette interconnexion des objets est rendue possible grâce à l'intégration de capteurs, de processeurs et de dispositifs de communication dans les objets physiques, leur permettant de collecter, de traiter et de partager des données en temps réel.

Notamment à **l’université Alioune Diop de Bambey**, le pointage des étudiants se fait par émargements de feuille. Cette méthode ancienne ne garantit pas l’authenticité des présences, est lente et est couteuse. Ainsi, pour solutionner, innover et digitaliser la méthode de pointage actuel, nous mettons en place un projet intitulé **système de pointage multifonctionnel par carte rfid et empreinte digitale**. Ce dernier est moderne, innovant, rapide, sécurisé et moins couteux.

Pour réaliser ce projet, nous l’avons scindé en 4 chapitres :

* Dans le premier chapitre, nous essayerons de faire la présentation générale,
* Le deuxième chapitre est destiné aux langages et outils de développement utilisés,
* Le troisième chapitre pour expliquer la conception détaillée du dispositif de pointage
* Le quatrième chapitre présente une évaluation du système proposé et les perspectives d’amélioration.

Chapitre I : Présentation générale :

## Présentation du sujet

### Contexte Générale

Avec l’ascension fulgurante de la digitalisation grâce à l’IoT qui permet de connecter des objets et des produits du quotidien dans un réseau numérique, permettant aux capteurs de recueillir des informations sur l'appareil, aux logiciels de surveiller ses performances et de collecter des données, et aux utilisateurs de le contrôler.

Avec l’essor de la **digitalisation** du système administratif public sénégalais, renforcé par l’arrivée du nouveau président de la république **son excellence Bassirou Diomaye Diakhar Faye** qui priorise la transformation digitale de l’administration sénégalaise pour moderniser et rendre plus efficaces les services publics. En effet, la digitalisation des données permet un accès plus facile aux informations, assure une optimisation du temps, d’argent en automatisant les taches de plus en plus complexes et la transmission des information plus fluide.

Ainsi plus particulièrement à l’**UABD,** son système de pointage par feuille se doit être moderne et digitalisé.

Ce dernier peut être rendu possible grâce à l'**IoT** qui a révolutionné de nombreux aspects de notre vie quotidienne avec des capteurs intelligents et à des appareils connectés.

### Problématique

L’usurpation d’identité entre étudiant lors des examens, le cout pour l’impression des feuilles de présence et la lenteur font partis des choses qui handicape le système de pointage par feuilles de l’UADB. C’est-à-dire un étudiant peut venir faire un exam à la place d’un autre. Elle est lente à cause des fil d’attente pour émarger après avoir rendu les copies, ou durant l’épreuve peut déconcentrer les étudiants et certains peuvent émarger sans rendre leurs copies.

Ainsi avec les cartes RFID et les systèmes de reconnaissances biométriques, le processus de pointage sera rendu plus rapide, plus précis et plus sécurisé.

### Objectifs

Ce projet vise à optimiser la gestion des présences et les ressources de l’université, à prévenir les fraudes académiques et à garantir une évaluation équitable pour tous les étudiants avec système de pointage facile à utiliser, rapide et sécurisé en suivant ces étapes :

1. Chaque carte étudiant sera associée à une puce rfid
2. On récupérera l’empreinte de chaque étudiant lors de l’inscription
3. Notre dispositif de pointage relier avec un site web va permet à l’administration avec leurs comptes utilisateurs de créer des événements pour une classe avec la durée de celui-ci, on peut créer plusieurs événements en même temps de classe différente et même des événements à venir
4. Lorsqu’un étudiant se pointe avec sa carte ou son empreinte, il est ajouté à l’historique de l’événement qu’il appartienne
5. Après chaque événement on retire les historiques sous formats Excel

## Présentation des technologies :

### Historiques de Arduino :

Arrivé sur le marché en 2005 comme un modeste outil pour les étudiants de l**’Interaction Design Institute Ivrea**, Arduino a initié une révolution dans l’univers de l’électronique à l’échelle mondiale. C’est une plate-forme électronique open source basée sur du matériel et des logiciels faciles à utiliser pour construire des projets électroniques. C’est un microcontrôleur qui est un circuit intégré capable de recevoir de l’information, qui va ensuite pouvoir commander d’autres composants électroniques. Arduino utilise une interface de programmation simplifiée. Il suffit d’un ordinateur d’un câble USB et d’un minimum de connaissance en programmation pour commencer à créer. Les cartes Arduino sont disponibles dans le commerce sous forme préassemblées ou en kits à monter soi-même illustrés par les deux figures suivant.



Figure I‑1 : Arduino

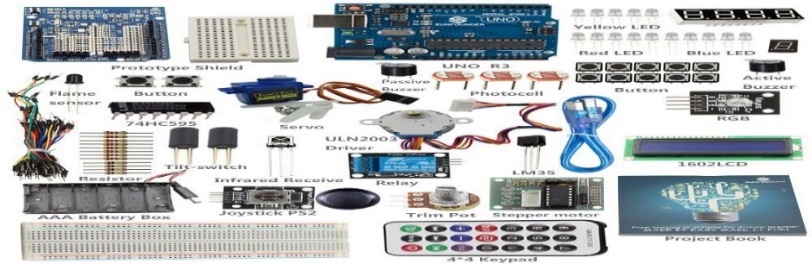


Figure 2 : Kit Arduino

### Les autres cartes :

Après avoir exploré les spécificités d'Arduino, il est crucial de se pencher sur d'autres plateformes de prototypage et d'embarqué qui ont également révolutionné le domaine des IOT tels que les cartes ESP et Raspberry.

#### ESP :

Les microcontrôleurs **ESP** sont développés par **Espressif Systems**, une entreprise basée en Chine fondée en 2008. Reconnus pour leur connectivité Wi-Fi intégrée et leur faible consommation d'énergie, les produits ESP, tels que l**'ESP8266** utiliser pour notre projet et **esp32** (les plus utilisés), sont particulièrement adaptés aux projets nécessitant une communication sans fil.

Voici une figure de l’esp8266 pour illustrer les microcontrôleurs ESP



Figure 3 : esp8266

#### Raspberry Pi :

Le Raspberry Pi est développé par **Raspberry Pi Foundation**, une organisation caritative basée au Royaume-Uni fondée en 2009. Se positionnant comme un mini-ordinateur polyvalent avec un système d'exploitation complet, le Raspberry Pi offre une puissance de traitement et une flexibilité supérieures pour des projets plus complexes nécessitant une gestion plus avancée des données et des interfaces utilisateur.

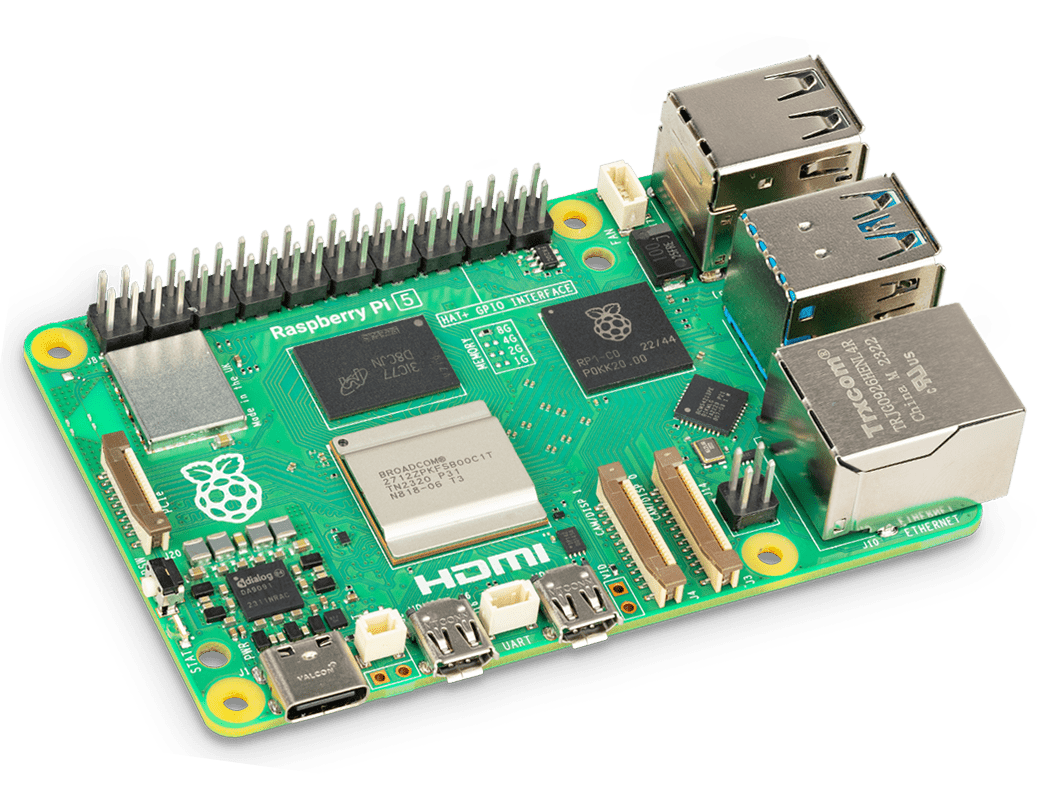


Figure 4 : image raspberry

En résumé, les cartes ESP et Raspberry Pi se distinguent par leurs architectures avancées, leurs capacités de connectivité et leurs performances, offrant ainsi des solutions alternatives pour des projets nécessitant des fonctionnalités plus spécifiques et des exigences techniques plus élevées illustrées par le tableau ci-dessous.

Tableau I-1 : Comparaison de l’Arduino, Esp et Raspberry Pi

|  |  |  |  |
| --- | --- | --- | --- |
| **Caractéristiques** | **Raspberry pi Pico** | **Arduino Nano** | **NodeMCU Esp8266** |
| Microcontrôleur | RP2040 | ATMega328P | ESP8266 |
| Core | Dual core | Single core | Single core |
| Architecture | 32 bit ARM cortex M0+ | 8 bit RISC | 32 bit LX106 |
| Fréquence | up to 133MHz | 16 MHz | Up to 160MHz |
| Tension | 3.3v | 5v | 3.3v |
| SPI/I2C/UART/I2s | 2/2/2 | 1/1/1 | 2/1/2/2 |
| Wifi | NO | NO | YES |
| Langage de  programmation | Micro python C C++ | Arduino IDE C/C++ | Arduino IDE C/C++ Micro Python JavaScript |
| Flash | 2MB | 32KB | 4MB |
| Ram | 264KB | 264KB | 128KB |

## Les outils utilisés :

### Esp8266

#### Caractéristique

L'ESP8266 est composé d'un microcontrôleur 32 bits (Tensilica Xtensa **LX106**) cadencé à 80MHz avec une mémoire Flash de 16 MO et une mémoire RAM de 32K + 80K. Il intègre la norme **Wi-Fi IEEE 802.11 b/g/n** avec possibilité d'authentification par mot de passe **WEP** ou **WPA**/**WPA2** avec une tension d’alimentation **3.3V** (Tolérance de tension : 3.0V - 3.6V), une tension logique (Niveau TTL) de 3.3V et une faible consommation d’énergie (**≤ 15 mA**). Il possède

* 16 broches entrées/sorties (**GPIO**), dont une entrée analogique 10 bits,
* Des pins d’alimentation tels que **VIN** (4V à 6V) **3V3** (3,3V) et **GND** (la masse),
* 2 broches pour la communication série (**RX** et **TX**) : Utilisées pour la communication série avec d'autres périphériques.
* 2 boutons reset (pour réinitialiser le programme) et flash (utilisé pour le processus de mise à jour du micrologiciel)
* Des broches spéciales pour les interfaces **SPI**, **I2C**, **SERIAL** etc.

La figure suivante illustre l’esp8266 avec ses broches :

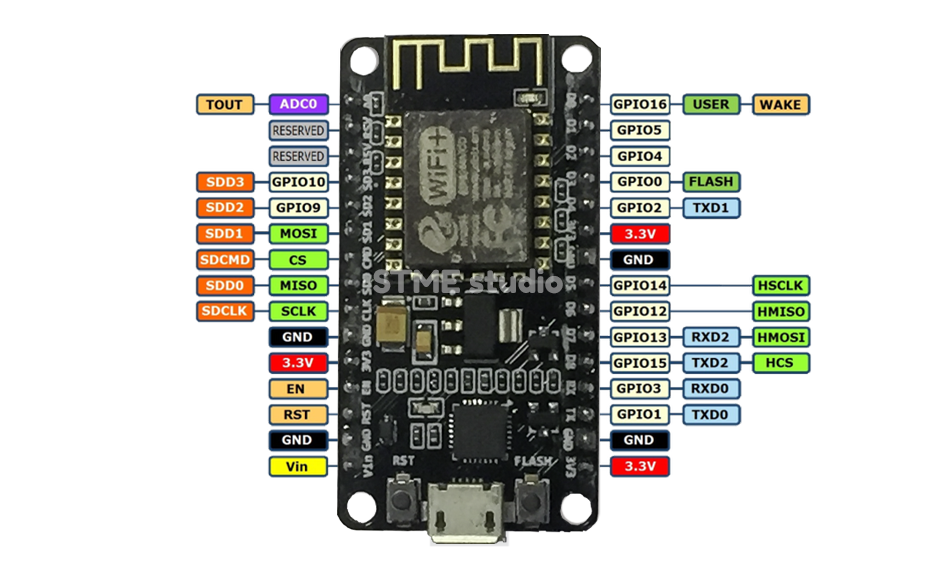


Figure 5 : L’illustration de l’esp8266 avec ses broches :

#### Rôle dans ce projet :

Dans ce projet, l'ESP8266 est le principal microcontrôleur et le dispositif de connectivité sans fil. Voici quelques rôles spécifiques de l'ESP8266 :

* **Connexion Wi-Fi :** L'ESP8266 est configuré en tant que station Wi-Fi pour se connecter au réseau Wi-Fi à l'aide des identifiants **SSID** et **mot de passe** spécifiés.
* **Communication avec le serveur :** Une fois connecté au réseau Wi-Fi, l'ESP8266 utilise la bibliothèque WiFiClient pour établir une connexion avec le serveur Django. Il envoie des données au serveur Django via des requêtes HTTP POST contenant les informations collectées par le module RFID et le capteur d'empreintes digitales.
* **Récupération des données de l'empreinte digitale et du RFID :** Lorsque des données sont disponibles à partir du lecteur RFID ou du capteur d'empreintes digitales, l'ESP8266 collecte ces données et les transmet au serveur.
* **Contrôle des périphériques de sortie :** L'ESP8266 contrôle également les périphériques de sortie, tels que le buzzer, en envoyant des signaux aux broches correspondantes pour indiquer des événements ou des actions réussies.

En résumé, l'ESP8266 joue un rôle central dans le projet en assurant la connectivité Wi-Fi, la lecture des capteurs, le traitement des données et la communication avec le serveur Django, tout en contrôlant les périphériques de sortie pour fournir des retours d'informations appropriés.

## RFID :

La technologie RFID repose sur l'utilisation d'ondes radio pour l'identification et la communication entre des dispositifs. Les cartes et les tags RFID sont équipés d'une puce électronique et d'une antenne qui leur permettent d'échanger des informations avec un lecteur RFID. Cette technologie trouve des applications dans divers secteurs, tels que la logistique, la sécurité, les paiements électroniques (paiements sans contact) et bien d'autres.



Figure 6 :RFID

### Cartes RFID

Les cartes RFID ressemblent à des cartes de crédit et sont souvent utilisées pour le contrôle d'accès, l'authentification et les paiements sans contact.

Les cartes RFID sont principalement classées en trois catégories : passives, semi-passives (ou semi-actives) et actives.

* Les **cartes passives** n'ont pas de batterie intégrée. Elles sont activées par l'énergie émise par le lecteur RFID et utilisent cette énergie pour transmettre les données stockées dans leur puce électronique.
* Les **cartes semi-passives** ont une batterie pour alimenter certaines fonctions internes, telles que la mémoire et les capteurs, mais elles utilisent toujours l'énergie du lecteur pour la communication.
* Les **cartes actives** ont leur propre source d'énergie (généralement une batterie) qui leur permet d'émettre un signal radio indépendamment du lecteur RFID, augmentant ainsi la portée et la fiabilité de la communication.



Figure 7 : La carte rfid

### Tags RFID

Le tag correspond à une étiquette électronique, appelée également transpondeur (Transmette -répondeur) ou marqueur. Comme les cartes, les tags RFID sont disponibles en versions passive, semi-passive et active, chacune avec des caractéristiques spécifiques :

* Les **tags passifs** sont légers, peu coûteux et peuvent être lus à des distances courtes à moyennes (de quelques centimètres à plusieurs mètres).
* Les **tags semi-passifs** ont une batterie pour alimenter des fonctions supplémentaires comme les capteurs de température, mais dépendent toujours de l'énergie du lecteur pour la communication.
* Les **tags actifs** ont une portée de lecture plus grande (pouvant atteindre plusieurs dizaines de mètres) et une capacité à transmettre des données à intervalles réguliers, indépendamment du lecteur RFID.



Figure 8 : tags rfid

### Lecteurs RFID

Les lecteurs sont des équipements actifs, portables ou fixes, qui sont constitués d’un circuit qui émet une énergie sous forme de champ magnétique ou d’onde radio. La communication entre le lecteur RFID et l'étiquette est possible grâce à chaque antenne RFID intégrée dans chacun des deux composants. Cette communication commence dès que le tag se trouve à une distance appropriée qui dépend des caractéristiques typiques des systèmes tels que la puissance ou la fréquence de travail.



Figure 9 : lecteur rfid

### Principe de fonctionnement

Le principe de fonctionnement est simple : le lecteur contient une bobine, lorsqu’elle est alimentée par une tension, elle génère un champ magnétique. Lorsque le Tag est assez proche du lecteur et baignant dans le champ magnétique, son antenne induit une tension dépendante du champ créé. Cette déférence de potentiel alimente la puce électronique. Elle utilise l’antenne ensuite pour changer les données avec le lecteur dans une courte distance par un nombre d’identification (UID) on peut dire donc que le tag RFID possède une mémoire qui comporte des octets pour stoker des informations.

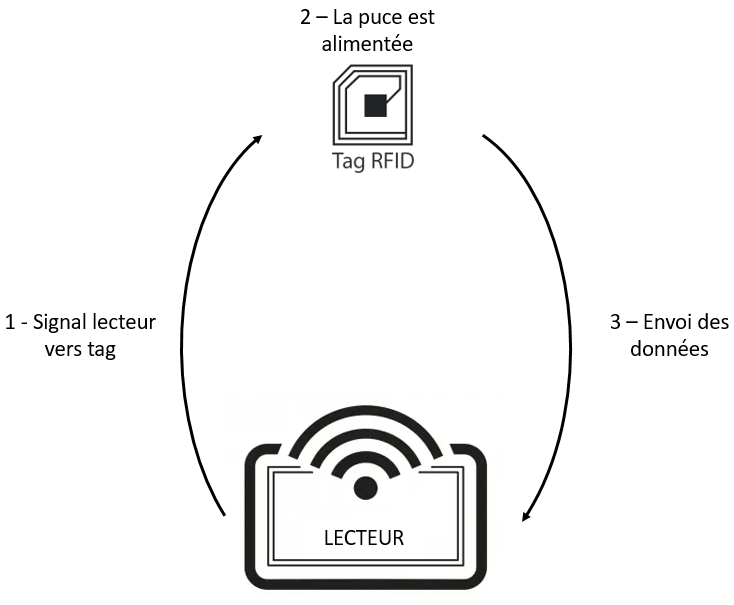


Figure 10 : principe de fonctionnement de rfid

## L’empreinte digitale R307S

R307S a la capacité de générer une image de haute qualité de l'empreinte pouvant contenir 1000 empreintes avec une interface communication de UART (TTL niveau logique), une tension de fonctionnement 3.6V à 6.0V, Courant de fonctionnement 100mA à 150mA, température de fonctionnement -20°C à +60°C.



Figure 11 : R307s

### Mécanisme de la capture d’empreinte

Le capteur R307 fonctionne en scannant l’empreinte digitale d’une personne et en créant une image numérique des motifs uniques de l’extrémité du doigt. Cette image est ensuite traitée et convertie en un algorithme mathématique qui est stocké dans le capteur. Lorsqu’une empreinte digitale est scannée à nouveau, le capteur compare la nouvelle image à l’algorithme précédemment stocké pour confirmer s’il y a correspondance. Si l’empreinte et l’algorithme sont identiques, le capteur envoie un signal indiquant que l’empreinte digitale est valide.

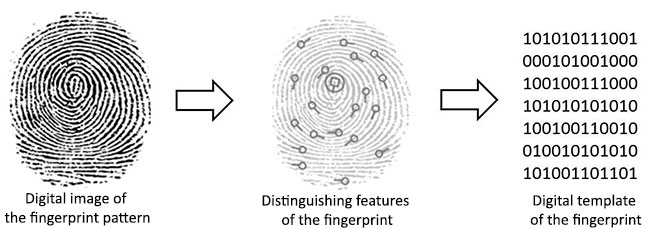


Figure 12 : Mécanisme de la capture d’empreinte

## La platine

Une platine d’expérimentation (appelée breadboard) permet de réaliser des prototypes de montages électroniques sans soudure et donc de pouvoir réutiliser les composants. Tous les connecteurs dans une rangée de 5 sont reliés entre eux. Donc si on branche deux éléments dans un groupe de cinq connecteurs, ils seront reliés entre eux. Il en est de même des alignements de connecteurs rouges (pour l’alimentation) et bleus (pour la terre). Ainsi, les liens peuvent être schématisés

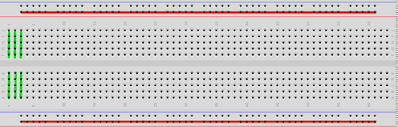


Figure 13 : la platine

## Le buzzer

Le piézo, également appelé buzzer, est un composant utilisé pour générer du son. Il s'agit d'un composant numérique qui peut être connecté aux sorties numériques et émet une tonalité lorsque la sortie est ÉLEVÉE. Alternativement, il peut être connecté à une sortie analogique de modulation de largeur d'impulsion pour générer diverses tonalités et effets.



# Chapitre II : Langages et outils de développement utilisés :

Le développement du système de pointage multifonctionnel a nécessité l'utilisation d'une variété de langages de programmation et d'outils de développement pour assurer la fonctionnalité, la robustesse, et la sécurité de la solution proposée.



## IDE Arduino

Le logiciel de programmation des modules Arduino, dont l'interface, appelée *Arduino IDE*, possède un compilateur qui va transformer votre programme en langage machine compréhensible par la carte Arduino.

L’Arduino IDE est un dérivé du logiciel Processing. C’est un logiciel Open Source et il peut être utilisé pour programmer des cartes autres qu’Arduino. Il est facile à utiliser pour les débutants, les Arduino sont plus faciles à apprendre à programmer car ils utilisent une version simplifiée du C++ par rapport aux autres logiciels de programmation qui nécessite une certaine connaissance. Ce langage de programmation simple est appelé Wiring, qui permet d’écrire des scripts qui ferra exécuter les taches par le microcontrôleur. Ces scripts sont appelés Sketches par Arduino.

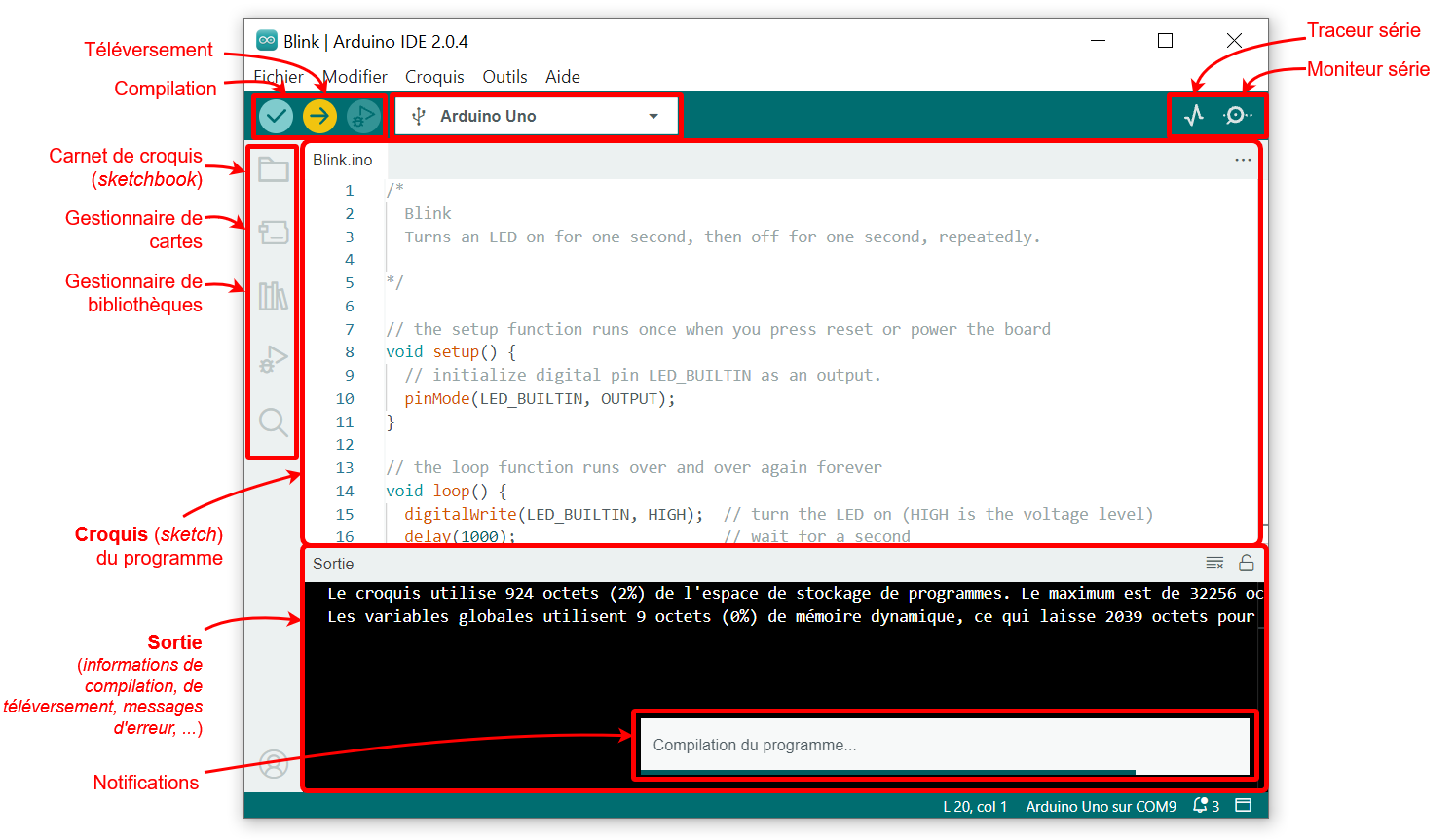


Figure 14 : : Illustration Arduino IDE

## Visual Studio Code

Visual Studio Code est un éditeur de code simplifié, qui est gratuit et développé en open source par Microsoft. Il fonctionne sous Windows, mac OS et Linux. Il fournit aux développeurs à la fois un environnement de développement intégré avec des outils permettant de faire avancer les projets techniques, de l’édition, à la construction, jusqu’au débogage.

Les fonctionnalités proposées par Visual Studio Code sont nombreuses. On retrouve notamment :

* **La prise en charge de plusieurs centaines de langage de programmation**, telles que C, C#, C++, CSS, HTML, Java, JavaScript, JSON, Markdown, PHP, PowerShell, Python, TypeScript, YAML…,
* **IntelliSense**, une fonction de complétion intelligente du code,
* **Un débogueur intégré**pour accélérer votre boucle d’édition, de compilation et de suppression des bugs,
* **Une interface d’édition**, qui intègre des raccourcis clavier, des sélections multiples, un enregistrement automatique de votre travail, une fonction rechercher/remplacer, le formatage du code source…,
* **Peek,** une fonction qui permet de parcourir rapidement le code source et de naviguer entre les fichiers,
* **Les commandes Git intégrées** ainsi que la gestion du contrôle des sources (SCM).

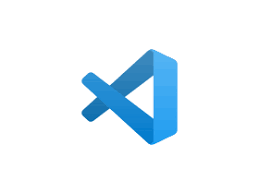


Figure 15 : logo Visual Studio Code

## Python

Python est le langage de programmation open source le plus employé par les informaticiens. Ce langage s’est propulsé en tête de la gestion d’infrastructure, d’analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu’ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu’avec d’autres langages. Ce puissant langage de programmation est utilisé par des milliers de sites dont les géants Google, Uber ou Reddit car sa flexibilité permet de créer des utilitaires web complexes avec une relative facilitée.

En ce qui concerne les frameworks, on peut utiliser Django (framework Python) pour faciliter le processus de gestion des fonctionnalités du backend ou du serveur.



Figure 16 :logo python

## Django

### Définition

Django est un framework Python de haut niveau, permettant un développement rapide de sites internet, sécurisés, et maintenables. Créé par des développeurs experimentés, Django prend en charge la plupart des tracas du développement web, vous pouvez donc vous concentrer sur l'écriture de votre application sans avoir besoin de réinventer la roue. Il est gratuit, open source, a une communauté active, une bonne documentation, et plusieurs options pour du support gratuit ou non.

**Django vous aide à écrire une application qui est :**

* **Polyvalent**

Django peut être (et a été) utilisé pour créer presque tous les genres de sites du gestionnaire de données aux wikis, jusqu'aux réseaux sociaux et aux sites d'actualités. Il peut fonctionner avec n'importe quelle infrastructure côté client, et peut renvoyer des données dans quasiment n'importe quel format (notamment HTML, RSS, JSON, XML, etc).

* **Maintenable**

Les principes de design du code Django encouragent la création d'un code simple à maintenir et réutilisable. Il fait notamment appel à la philosophie du Ne Vous Répétez Pas (DRY pour Don't Repeat Yourself en anglais), afin d'éviter toute duplication superflue, réduisant la taille de votre code. Django promeut aussi le regroupement de fonctionnalités reliées entre elles en "applications" réutilisables et, à un plus bas niveau, regroupe des lignes de code dépendantes entre elles en modules (suivant les lignes du motif d'architecture Modèle-vue-contrôleur (MVC)).

* **Sécurisé**

Django aide les développeurs à éviter les erreurs de sécurité classique en fournissant une infrastructure conçue pour "faire ce qu'il faut" pour protéger les sites internet automatiquement. Par exemple, Django fournit un moyen sécurisé pour gérer les comptes des utilisateurs ainsi que leurs mots de passe, évitant les erreurs classiques comme mettre des informations sur la session dans des cookies, où elles sont vulnérables (à la place les cookies contiennent seulement une clé, et les données sont stockées dans la base de données), ou directement stocker des mots de passe, au lieu de mot de passe hachés.

Django active par défaut la protection contre beaucoup de vulnérabilités, comme les injections SQL, le cross-site scripting, le cross-site request forgery et le clickjacking

* **Portable**

Django est écrit en Python, qui fonctionne sous diverses plateformes. Cela veut dire que vous ne serez plus contraint par une plateforme en particulier, et vous pourrez faire fonctionner vos applications sous autant de versions de Linux, Windows et Mac OS X que vous le souhaitez. De plus, Django est très bien supporté par plusieurs fournisseurs d'hébergement web, qui offrent souvent des infrastructures et de la documentation spécifique pour héberger des sites Django.



Figure 17 :logo Django

### Fonctionnement

Les applications web Django regroupent généralement le code qui gère chacune de ces étapes dans des fichiers séparés :



Figure 18 : fonctionnement de django

* **URLs :**

Bien qu'il soit possible de traiter les requêtes de chaque URL via une fonction unique, il est bien plus viable d'écrire une fonction de vue isolée qui gèrera chaque ressource. Un mapper URL est utilisé pour rediriger les requêtes HTTP à la vue appropriée d'après l'URL de requête. Le mapper URL peut aussi faire la correspondance entre des patterns de chaînes de caractères ou de nombres qui apparaissent dans une URL et passer ces derniers comme données dans une fonction de vue.

* **Vues :**

Une vue est une fonction de gestion des requêtes, qui reçoit des requêtes HTTP et renvoie des réponses HTTP. Les vues accèdent aux données requises pour satisfaire des requêtes via des *modèles*, et délèguent le formatage des réponses aux *templates*.

* **Modèles :**

Les modèles sont des objets Python, qui définissent la structure des données d'une application, et fournissent des mécanismes de gestion (ajout, modification, suppression) et requêtent les enregistrements d'une base de données.

* **Templates :**

Un template est un fichier texte qui définit la structure ou la mise en page d'un fichier (comme une page HTML), avec des balises utilisées pour représenter le contenu. Une *vue* peut créer une page HTML en dynamique en utilisant un template HTML, en la peuplant avec les données d'un *modèle*.

Figure 19 : logo Font Awesome

## Django base donnée et ORM

* **Base de donnée**

Django, un framework Web de haut niveau pour Python, inclut une prise en charge intégrée pour l'utilisation de SQLite comme backend de base de données. SQLite est un moteur de base de données léger et sans serveur qui stocke les données dans un fichier local. C'est un bon choix pour le développement et les applications de petite et moyenne taille.

* **ORM**

Au cœur de Django se trouve son système ORM qui fournit un pont entre la base de données relationnelle et le code Python, facilitant une interaction transparente avec la base de données sans avoir besoin d'écrire manuellement des requêtes SQL. Il traduit le code Python en requêtes SQL en arrière-plan, rendant les interactions avec les bases de données plus intuitives et moins sujettes aux erreurs.

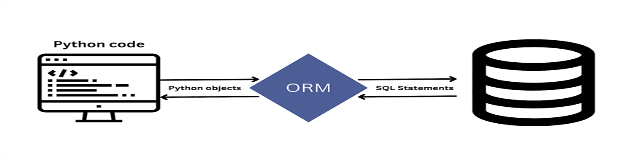


Figure 20 : illustration de ORM

## HTML

Pour créer des sites Web, vous devez connaître le HTML technique fondamentale utilisée pour définir la structure d'une page Web. HTML est utilisé pour dire si votre contenu web doit être reconnu en tant que paragraphe, liste, en-tête, lien, image, lecteur multimédia, formulaire ou l'un des nombreux autres éléments disponibles ou même un nouvel élément à définir par vous-même.



Figure 21 : logo HTML

## CSS

Css est un langage informatique de description utilisé pour compléter le HTML, et lui conférer un style personnalisé et personnalisable. Il permet d’attribuer un design spécifique à tout élément du HTML, lui modifiant ses caractéristiques par défaut. Grâce à lui, il est possible de changer des couleurs de titre, encadrer un contenu, attribuer une taille spécifique à un bloc, et pléthore d’autres manipulations allant même jusqu’à l’animation de contenu.



Figure 22 : logo css

## JavaScript

JavaScript est un langage de programmation utilisé par les développeurs pour concevoir des sites web interactifs. Les fonctions JavaScript peuvent permettre d'améliorer l'expérience utilisateur d'un site web, de la mise à jour des flux de médias sociaux à l'affichage d'animations et de cartes interactives. En tant que langage de script côté client, c'est l'une des principales technologies du web. Lors de la navigation sur Internet, à tout moment vous pouvez par exemple voir un carrousel d'images, un menu déroulant « Cliquer pour afficher » ou le changement dynamique de la couleur des éléments d'une page web. Tout cela est possible grâce à JavaScript.

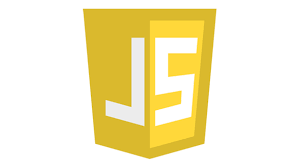


Figure 23 : logo JavaScript

## Bootstrap

Bootstrap est un framework CSS open-source développé par Twitter. Il fait partie de ces Front-End Framework contenant une collection d’outils utiles à la création et au design de sites et d’applications web en front end. Il contient des codes HTML et CSS ainsi que des extensions JavaScript en option, permettant la création de formulaires, de boutons, d’outils de navigations et d’autres éléments interactifs à intégrer dans une application. Bootstrap est facile à utiliser, n’importe qui ayant une connaissance basique des langages HTML et CSS peut le comprendre. De plus, ce framework correspond à **une solution responsive**qui s’ajuste aux téléphones, aux tablettes et aux ordinateurs.



Figure 24 : logo Bootstrap

## Font Awesome

Font Awesome est une police de caractères qui permet d’afficher des icônes, des pictogrammes sur un site web. Elle contient une collection d'icônes prêtes à être utilisées sous forme de police ou de fichiers SVG, ce qui permet aux développeurs d'ajouter facilement des icônes stylisées à leurs sites web et applications. Font Awesome est populaire en raison de sa facilité d'utilisation, de sa diversité d'icônes et de sa compatibilité avec de nombreux frameworks et outils de développement.



Figure 25 : font awesome

Haut du formulaire

## UML

UML est un langage de modélisation standardisé utilisé pour représenter visuellement les différents aspects d'un système logiciel, y compris sa structure, son comportement, et ses interactions. Il regroupe plusieurs sous-catégories de diagrammes, notamment diagramme des cas d'utilisation, diagramme de séquence UML, diagramme des composants UML, diagramme de classe UML

* **Diagramme UML des cas d'utilisation**

Un diagramme de cas d'utilisation est un type de diagramme utilisé en UML pendant la phase d'analyse d'un projet pour identifier la fonctionnalité du système. Il décrit l'interaction des personnes ou du dispositif externe avec le système en cours de conception. Il ne montre pas beaucoup de détails, mais résume seulement certaines des relations entre les cas d'utilisation, les acteurs et les systèmes.

* **Diagramme de séquence UML**

Un diagramme de séquence est utilisé en UML pour décrire les phases d'analyse et de conception. C'est un diagramme d'interaction qui détaille la manière dont les opérations sont effectuées. Un diagramme de séquence est souvent utilisé pour représenter le déroulement chronologique des événements à travers un cas d'utilisation. Il permet de présenter les relations de communication entre les objets et les messages qui déclenchent ces communications.

* **Diagramme des composants UML**

Il permet aux concepteurs d'applications de vérifier que les fonctionnalités requises d'un système sont mises en œuvre par les composants, garantissant ainsi que le système final sera acceptable. De plus, le diagramme des composants est un outil de communication utile entre les parties prenantes pour discuter, analyser ou améliorer la conception du système.

* **Diagramme de classe UML**

Un diagramme de classe UML est non seulement utilisé pour décrire les structures des objets et des informations dans une application, mais aussi pour montrer la communication avec ses utilisateurs. Il offre un large éventail d'utilisations, de la modélisation de la vue statique d'une application à la description des responsabilités d'un système. La composition est un type spécial d'agrégation qui dénote une forte propriété.



Figure 26 : illustration UML

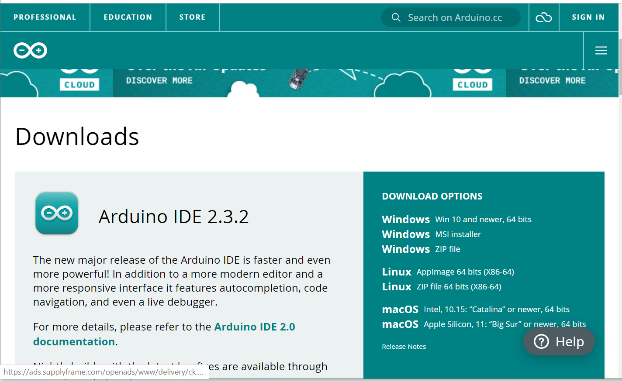
# Chapitre III : Conception détaillée du dispositif de pointage



## Architecture générale

## CONFIGURATION DE ESP8266 POUR ARDUINO :

Pour télécharger le logiciel, il faut se rendre sur la page de téléchargement du site arduino.cc <https://www.arduino.cc/en/software> . Vous avez deux catégories : Download : Dans cette catégorie, vous pouvez télécharger la dernière version du logiciel. Les plateformes Windows, Linux et Mac sont supportées par le logiciel.

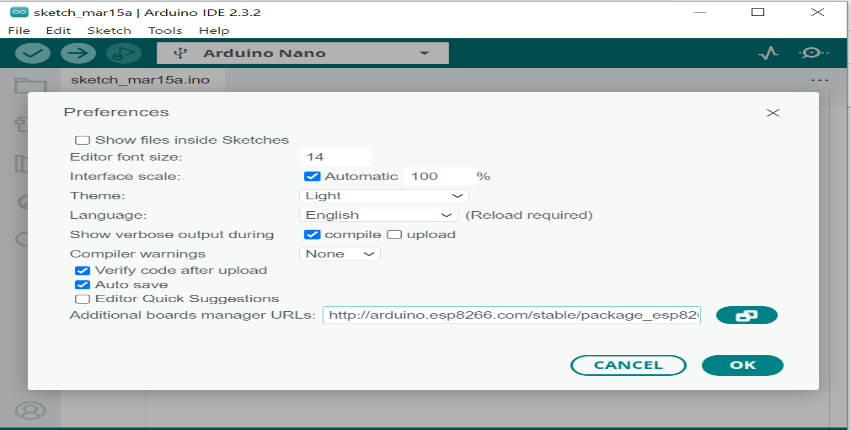


**Étape n°1 :**

La première étape est la configuration de l’IDE Arduino pour la prise en compte de la carte Esp8266.

* Ouvrir l’IDE Arduino.
* Ensuite :
  + File*->*Préférences
  + Dans la boite de dialogue des préférences, au champ intitulé “URL de gestionnaire de cartes supplémentaires“, vous pouvez coller l’adresse suivante :

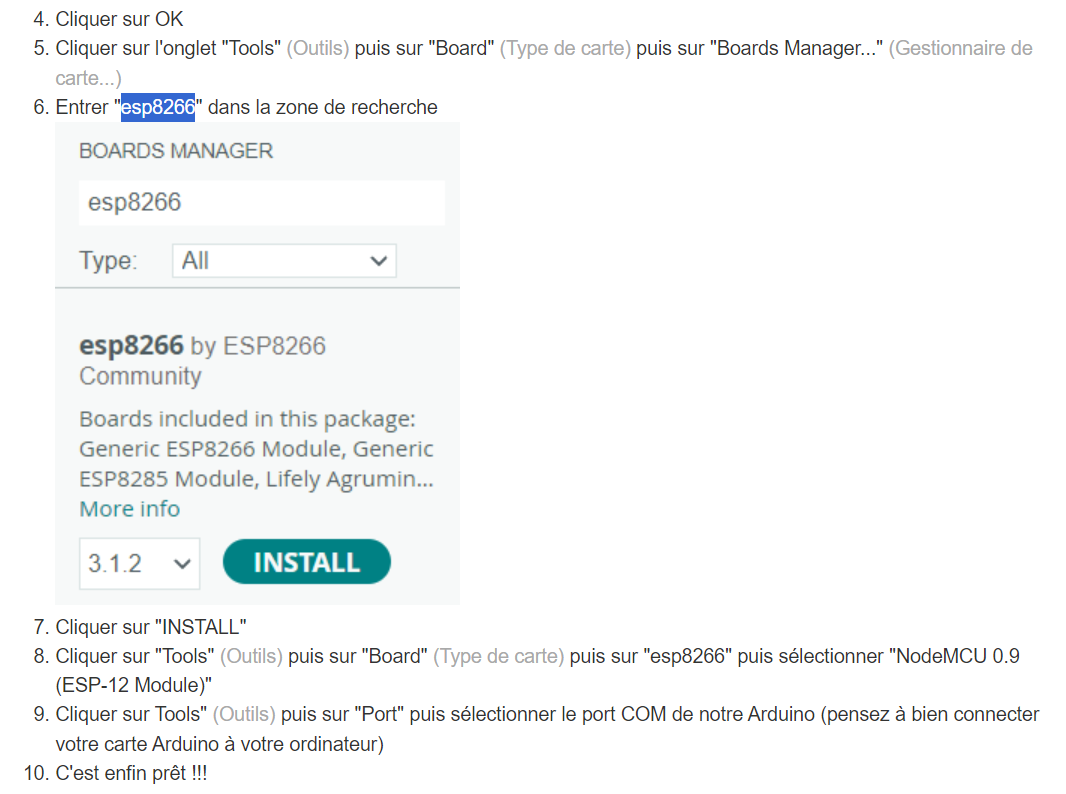
[**http://arduino.esp8266.com/stable/package\_esp8266com\_index.json**](http://arduino.esp8266.com/stable/package_esp8266com_index.json)



* Validez cette modification avec le bouton OK.

**Étape n°2 :**

Dans le menu : Tools->Board -> Board Manager



**Étape n°3 :**

Après quelque minutes d’installation, vous pouvez sélectionner la nouvelle carte dans :

* *Tools->Board->esp8266->NodeMCU 1.0(ESP-12E Module)*



**Étape n°4 :**

Branchez la carte au port USB du PC

• Télécharger le pilote sur ce lien <https://fr.drvhub.net/devices/input-devices/usb/serial-ch340>.

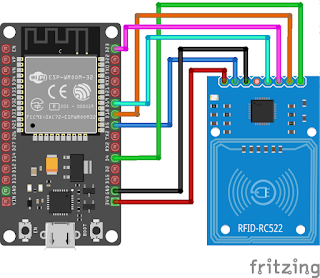
Exécutez le programme d'installation. Sur le Gestionnaire de périphériques de votre pc, développez Ports, vous pouvez trouver le port COM pour CH340.

Dans IDE Arduino, sélectionnez le port COM avec la carte.

## Branchement des composants

### Esp8266 et rfid

Le module RFID RC522 utilise la protocole SPI pour communiquer avec l’ESP8266. La communication SPI utilise des boches spécifiques sur ce type de microcontrôleur.



Le brochage se fait comme suit :

**Rfid** -------------------------> **Esp8266**

SDA -------------------------> D8

SCK -------------------------> D5

MOSI -------------------------> D7

MISO -------------------------> D6

IRQ -------------------------> Non connecté

GND -------------------------> GND

RST -------------------------> D3

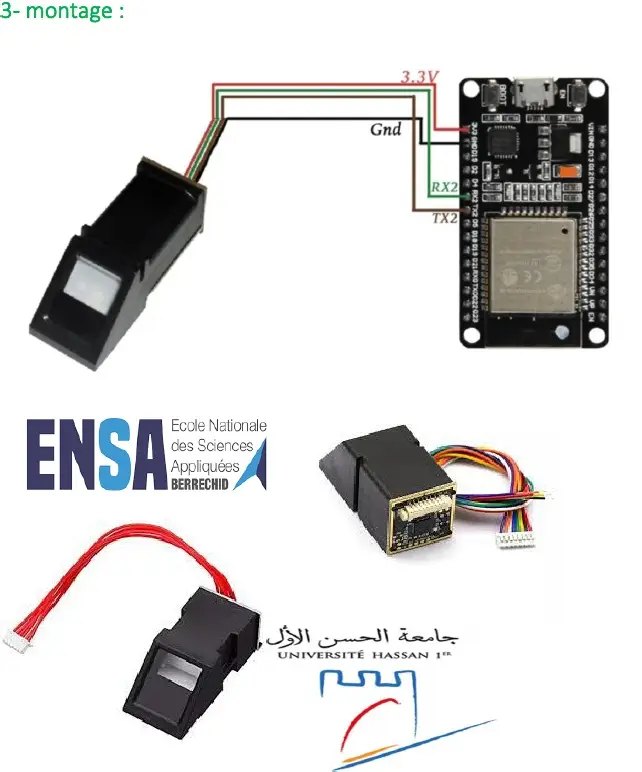
3V3 -------------------------> 3V3

Figure 27 :branchement esp8266 et rfid

### Esp8266 et R307s

R307S établit une communication série UART entre le module ESP8266

Le brochage se fait comme suit :



**R307s** -------------------------> **Esp8266**

3,3V -------------------------> 3,3V

GND -------------------------> GND

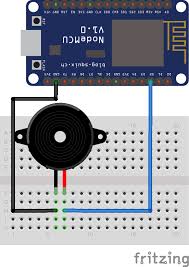
RX -------------------------> RX

TX -------------------------> D4

Figure 28 : branchement esp8266 et r307s

### Esp8266 et buzzer

Lorsque le lecteur rfid détecte une carte ou le capteur détecté une empreinte le buzzer sonne

**Buzzer** -------------------------> **Esp8266**

Coté court (-) -------------------------> GND

Coté long (+) -------------------------> D1

Figure 29 : Branchement esp8266 et le buzzer

### Branchement de tous les composants

## Présentation de la page web

Webographie :

* <https://reso-nance.org/wiki/materiel/esp8266/accueil>
* https://developer.mozilla.org/fr/docs/Learn/Server-side/Django/Introduction
* https://www.mrelectrouino.com/2021/01/Raspberry%20pi%20pico%20compare%20with%20arduino%20uno%20esp32%20esp8266.html
* <https://sbedirect.com/fr/blog/article/comprendre-la-rfid-en-10-points.html>
* <https://www.rfiddiscovery.com/en/content/passive-rfid>
* https://diyprojectslab.com/r307-fingerprint-sensor-with-arduino/
* <https://easypartner.fr/blog/arduino-la-carte-de-programmation-qui-fait-sa-revolution/#:~:text=Arduino%20a%20%C3%A9t%C3%A9%20construit%20autour,r%C3%A9guli%C3%A8rement%20pour%20%C3%A9tancher%20sa%20soif>
* <https://fr.macerobotics.com/developpeur/tutoriels/programmer-le-microcontroleur-esp8266-avec-lide-arduino>
* <https://subscription.packtpub.com/book/iot-and-hardware/9781787288102/1/ch01lvl1sec14/connecting-the-esp8266-to-your-local-wi-fi-network>