

Résume :

L'Identification par Radio Fréquence est une technologie très prometteuse dans tous les domaines. Dans ce travail nous avons conçu et réalisé un système de gestion de présence exploitant la technologie RFID. Il est constitué de deux parties majeures à savoir l'électronique et l'informatique. La partie informatique comprend une base de données et une interface d'utilisateur web et le développement d'un système d'exploitation en se basant sur les technologies Linux. Alors que la partie électronique contient principalement des modules wifi, RFID et un contrôleur.

Sommaire :

Résume :	3
Introduction :	5
Chapitre1 : Analyse et conception :	8
Diagramme de cas d'utilisation :	8
Diagramme de séquence :	8
Diagramme de classe :	9
Conclusion :	11
Chapitre2 : Conception technique :	13
Les outils électroniques :	13
Une carte Arduino :	13
Module RFID :	13
Afficheur LCD I2C :	14
Le module ESP8266 :	14
Raspberry Pi	15
Les outils informatiques :	16
Go :	16
PostgreSQL.....	16
L'IDE Arduino	16
Chapitre3 : Présentation du projet	19
Création de la base de données :	19
PostgreSQL.	19
Programmation de microcontrôleur.....	19
Fonctionnement de système	22
Les Interfaces :	23
Conclusion et Perspectives :	26

Introduction :

Les technologies sont présentes dans nos vies et elles ne cesseront de nous étonner par la qualité et la rapidité de ses fonctions. Elles améliorent nos vies par sa quantité d'informations ouvertes à tous, elles informent de tout ce qui se produit partout dans le monde, elles nous dirigent dans la prise de décision et permettent de s'actualiser facilement. En éducation, il a toujours été une préoccupation pour l'homme d'identifier, de localiser et de suivre des objets en utilisant d'abord l'identification visuelle puis remplacée par des équipements électroniques. Plusieurs systèmes pratiques ont été utilisés au cours des années, des motifs uniques ont été placés sur des objets, et des appareils de reconnaissances pouvaient identifier ces codes et par la même voie l'objet sur lequel ils sont collés. De là est né le système de codes à barres qui, pendant plusieurs années, a permis de réaliser ce rêve d'identification des objets. Cependant, les codes à barres présentent plusieurs lacunes, notamment le manque de stockage de données, la nécessité de les scanner à une distance de quelques mètres...etc. Ces déficits ont continuellement poussé l'homme à la recherche d'une meilleure solution pour pallier ce manque, et voilà pourquoi est née la technologie RFID qui à priori résolvait les majeurs problèmes d'identification, de localisation, de suivi et d'analyse de données.

Nous vous proposons également un projet de réalisation d'un système embarqué pour le pointage d'absence, en utilisant Arduino, un afficheur LCD I2C et un module RFID. Pour rendre le système connecté, nous avons choisi d'utiliser un module ESP8266 pour le wifi. Le système d'exploitation pour l'embarqué que nous proposons est conçu pour offrir une grande flexibilité et une grande modularité, afin de permettre une utilisation optimale des ressources limitées des microcontrôleurs et des microprocesseurs. Nous avons développé ce système d'exploitation en nous basant sur les technologies Linux, qui sont robustes et largement utilisées dans l'industrie. Pour le projet de pointage d'absence, nous avons utilisé Arduino, qui est un microcontrôleur

très populaire et facile à utiliser. Nous avons également utilisé un afficheur LCD I2C pour afficher les informations du système et un module RFID pour identifier les utilisateurs du système. Nous avons ensuite connecté le système à Internet en utilisant un module ESP8266, ce qui permet aux utilisateurs d'accéder aux données de pointage à distance. Nous espérons que notre site sera une ressource précieuse pour les développeurs et les amateurs intéressés par les systèmes d'exploitation pour l'embarqué, ainsi que pour les projets d'automatisation et de connectivité.

Chapitre 1

Analyse et conception

La phase de la conception est la phase initiale de la création et de la mise en œuvre de notre projet. En fait, elle représente une étape importante de réflexion dans le cycle de développement logiciel après la phase de l'analyse et de spécification.

Dans cette partie, nous allons présenter en détails la conception du projet à travers les diagrammes UML suivants : les diagrammes des cas d'utilisations, les diagrammes de séquence, le diagramme de classes

Chapitre1 : Analyse et conception :

Diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation décrit les utilisations requises d'un système, ou ce qu'un système est supposé faire. Les principaux concepts de ce diagramme sont les acteurs, cas d'utilisation et sujets. Un sujet représente un système avec lequel les acteurs et les autres sujets interagissent.

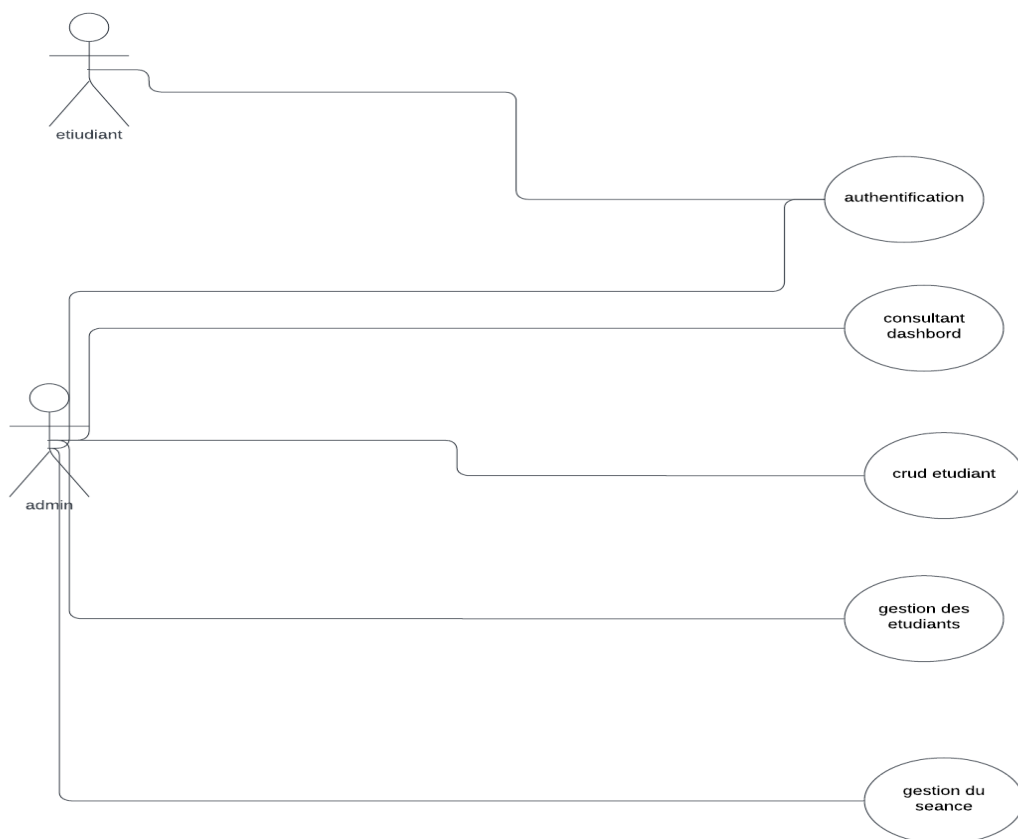


Diagramme de séquence :

Un diagramme de séquence est un document graphique qui montre pour des scénarios de cas d'utilisation précis, les événements générés et les interactions entre objets en se basant sur des messages ordonnés. Chaque message transitant sur un lien est symbolisé par une flèche porteuse d'une expression. La lecture se

fait de haut en bas, et l'ordre chronologique doit respecter ce sens. La réalisation de diagramme de séquence permet de lister les méthodes dont on aura besoin lors de la phase de développement. Pour ce faire la description doit être suffisamment générale et exhaustive pour identifier tous les algorithmes.

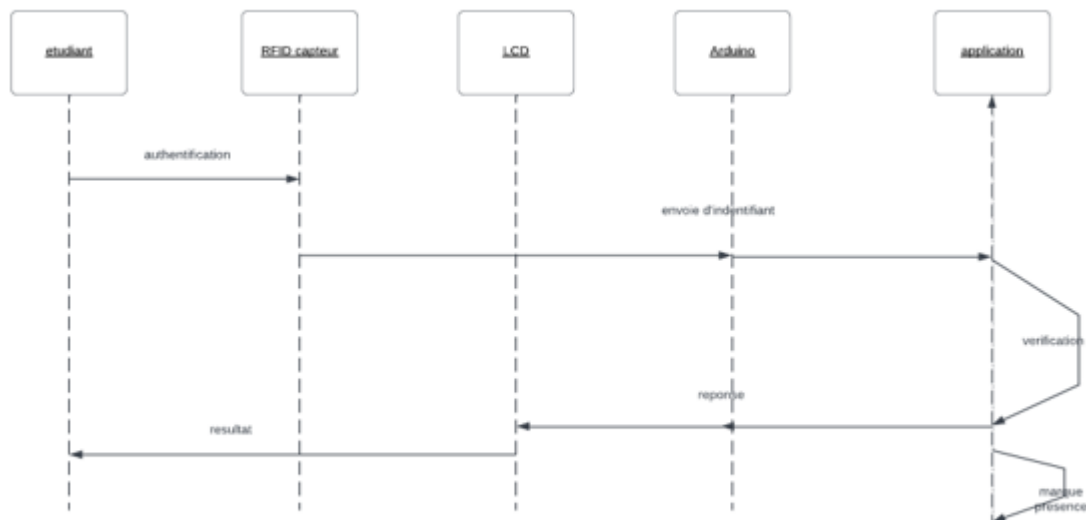


Diagramme de classe :

Le diagramme de classe est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.

Le diagramme de classe montre la structure interne de système. Il permet de fournir une représentation abstraite des objets du système qui vont interagir ensemble pour réaliser les cas d'utilisation. Il s'agit d'une vue statique car on ne tient pas compte du facteur temporel dans le comportement du system.

Les principaux éléments de cette vue statique sont les classes et leurs relation: association, généralisation et plusieurs types de dépendances, telles que la réalisation et l'utilisation.

Une classe-association possède les caractéristiques des associations et des classes : elle se connecte à deux ou plusieurs classes et possède également des attributs et des opérations. Une classe-association est caractérisée par

un trait discontinu entre la classe et l'association.

Une classe est une description d'un groupe d'objets partageant un ensemble

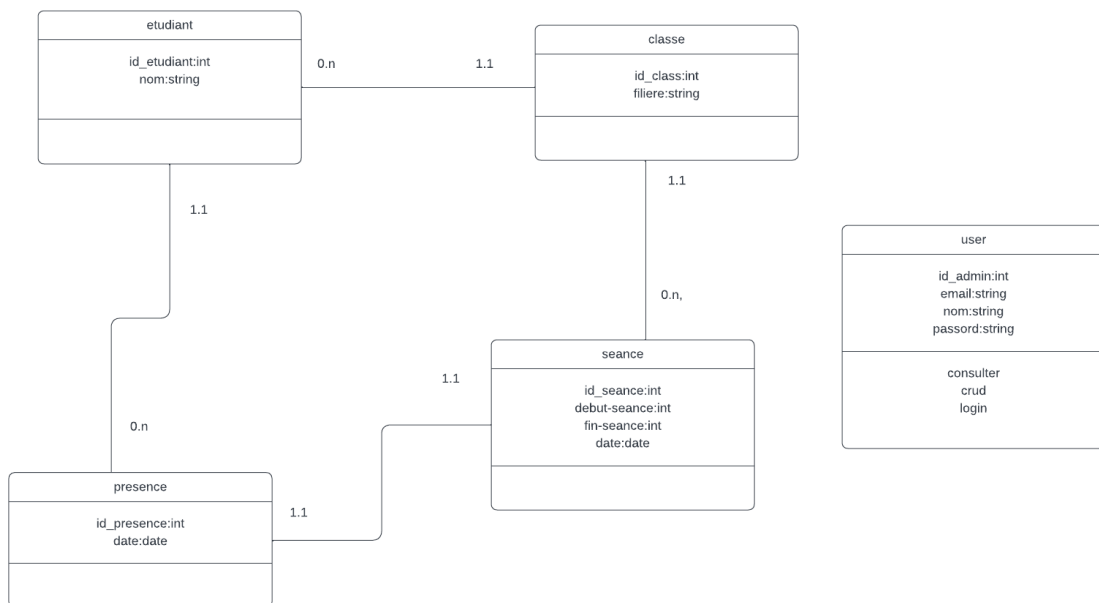
commun de propriétés (les attributs), de comportements (les opérations ou méthodes) et de relations avec d'autres objets (les associations et les agrégations).

Une classe de conception est composée par :

➤ Attribut chaque attribut d'une classe est le même pour chaque instance de cette

classe.

➤ Méthodes elle définit le comportement d'une classe elle-même, et non le comportement de ses instances qui peut être différent.



Conclusion :

La phase conceptuelle est une étape fondamentale pour la réalisation de n'importe quel projet. Elle permet de faciliter le système d'information et réaliser l'implémentation de la base des données et le traitement. Par la suite, on doit chercher les moyens et les outils possibles pour développer cette application, ce qu'on va présenter dans le chapitre suivant.

Chapitre2

Conception technique

Ce chapitre est consacré à la présentation de l'environnement logiciel utilisés pour le développement de la solution propose, nous expliquerons éventuellement nos choix techniques relatif aux langages de programmation et des outils utilisés

Chapitre2 : Conception technique :

Les outils électroniques :

Une carte Arduino :

Une carte Arduino est une carte électronique programmable open-source basée sur un microcontrôleur. Elle est utilisée pour créer des projets électroniques interactifs, tels que des robots, des capteurs et des dispositifs domotiques, en combinant des entrées et des sorties numériques et analogiques avec des instructions programmées. La carte Arduino est populaire pour sa facilité d'utilisation, sa flexibilité et son faible coût. Elle peut être programmée en utilisant un environnement de développement intégré (IDE) facile à utiliser, qui permet aux développeurs de créer et de téléverser du code sur la carte.



Module RFID :

Un module RFID est un circuit électronique qui intègre un ensemble de composants nécessaires pour réaliser une fonctionnalité RFID. Les modules RFID sont généralement conçus pour être faciles à intégrer dans des projets électroniques ou informatiques, et ils sont souvent accompagnés d'une documentation détaillée pour aider les développeurs à les utiliser. Les modules RFID peuvent être utilisés pour lire et écrire des données sur des étiquettes ou des cartes RFID, pour communiquer avec des lecteurs RFID externes, ou pour créer des réseaux de communication sans fil entre les dispositifs RFID. Les modules RFID sont largement utilisés dans les applications telles que la gestion des stocks, la logistique, l'identification des personnes, les systèmes de sécurité et de contrôle d'accès, etc.



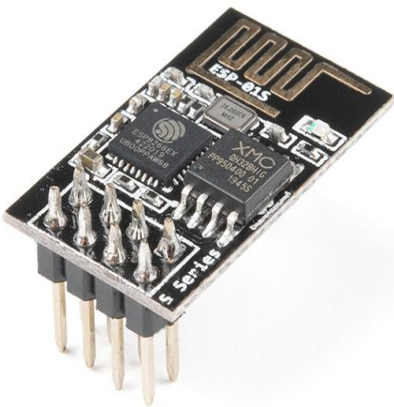
Afficheur LCD I2C :

Un afficheur LCD I2C est un écran à cristaux liquides (LCD) qui utilise le protocole de communication I2C pour échanger des données avec un microcontrôleur ou un autre dispositif électronique. Le protocole I2C permet une communication bidirectionnelle entre le microcontrôleur et l'afficheur LCD à l'aide de seulement deux fils de données (SCL et SDA), ce qui facilite l'intégration de l'afficheur LCD dans des projets électroniques ou informatiques. L'afficheur LCD I2C est souvent utilisé pour afficher des informations en texte ou en graphique dans des projets électroniques tels que des systèmes de mesure, des dispositifs de contrôle, des alarmes, des capteurs, etc. L'utilisation de l'interface I2C permet également de contrôler plusieurs afficheurs LCD à partir d'un seul microcontrôleur, ce qui permet d'économiser des ressources matérielles et de simplifier le développement de projets plus complexes.



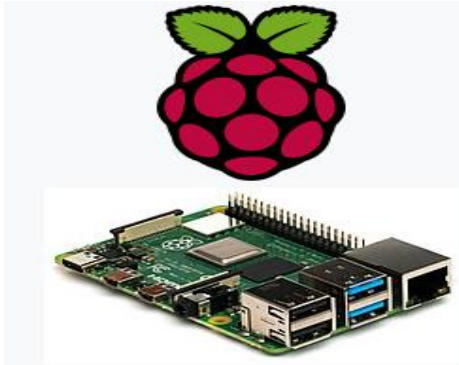
Le module ESP8266 :

Le module ESP8266 est un petit circuit électronique qui intègre un microcontrôleur et un module WiFi dans un seul et même package. Ce module est conçu pour offrir une solution de connectivité Wi-Fi à faible coût et à faible consommation d'énergie pour les projets électroniques ou informatiques. Le microcontrôleur intégré est basé sur une architecture à 32 bits et peut être programmé avec le langage de programmation Arduino ou avec le langage de script Lua. Le module WiFi prend en charge les protocoles 802.11 b/g/n et peut être configuré en mode client ou en mode point d'accès. Le module ESP8266 est largement utilisé pour ajouter des fonctionnalités de connectivité Wi-Fi à des projets électroniques tels que des capteurs, des systèmes de mesure, des dispositifs de contrôle, des objets connectés (IoT), etc. Il est également souvent utilisé pour le développement de projets de domotique ou de surveillance à distance.



Raspberry Pi

Raspberry Pi est une série de petits ordinateurs monocarte (SBC) développés au Royaume-Uni par la Raspberry Pi Foundation en association avec Broadcom. Le projet Raspberry Pi



était à l'origine orienté vers la promotion de l'enseignement de l'informatique de base dans les écoles. Le modèle original est devenu plus populaire que prévu, se vendant en dehors de son marché cible pour des utilisations telles que la robotique. Il est largement utilisé dans de nombreux domaines, tels que la surveillance météorologique, en raison de son faible coût, de sa modularité et de sa conception ouverte. Il est généralement utilisé par les amateurs d'informatique et d'électronique, en raison de son

adoption des normes HDMI et USB. Après la sortie du deuxième type de carte, la Fondation Raspberry Pi a créé une nouvelle entité, nommée Raspberry Pi Trading, et a installé Eben Upton en tant que PDG, avec la responsabilité de développer la technologie. La Fondation a été reconvertie en tant qu'organisme de bienfaisance éducatif pour promouvoir l'enseignement de l'informatique de base dans les écoles et les pays en développement. La plupart des Pis sont fabriqués dans une usine Sony à Pencoed, au Pays de Galles, tandis que d'autres sont fabriqués en Chine et au Japon. En 2015, le Raspberry Pi a dépassé le ZX Spectrum en ventes unitaires, devenant l'ordinateur britannique le plus vendu.

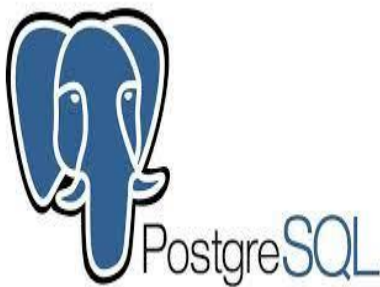
Les outils informatiques :

Go :



Go est un langage de programmation compilé et concurrent inspiré de C et Pascal. Ce langage a été développé par Google⁶ à partir d'un concept initial de Robert Griesemer (en), Rob Pike et Ken Thompson. Go possède deux implémentations : la première utilise gc, le compilateur Go ; la seconde utilise gccgo, « frontend » GCC écrit en C++. Go est écrit en C en utilisant yacc et GNU Bison pour l'analyse syntaxique jusqu'à la version 1.4.

PostgreSQL



PostgreSQL est un système de gestion de base de données relationnelle et objet (SGBDRO). C'est un outil libre disponible selon les termes d'une licence de type BSD.

Ce système est comparable à d'autres systèmes de gestion de base de données, qu'ils soient libres (comme MariaDB et Firebird), ou propriétaires (comme Oracle, MySQL, Sybase, DB2, Informix et Microsoft SQL Server). Comme les projets libres Apache et Linux, PostgreSQL n'est pas contrôlé par une seule entreprise, mais est fondé sur une communauté mondiale de développeurs et d'entreprises.

L'IDE Arduino



Le logiciel Arduino est un Environnement de Développement Intégré (IDE) open source et gratuit, téléchargeable sur le site officiel Arduino.

L'IDE Arduino permet : d'éditer un programme : des croquis (*sketch* en Anglais), les programmes sont écrits en langage C, de compiler ce programme dans le langage « machine » de l'Arduino,

la compilation est une traduction du langage C vers le langage du microcontrôleur la console donne des information sur le déroulement de la compilation et affiche les messages d'erreur. De téléverser le programme dans la mémoire de l'Arduino, le téléversement (upload) se passe via le port USB de l'ordinateur un fois dans la mémoire de l'Arduino, le logiciel s'appelle un microgiciel. La console donne des informations sur le déroulement du téléversement et affiche les messages d'erreur. De communiquer avec la carte Arduino grâce au terminal (ou moniteur série). pendant le fonctionnement du programme en mémoire sur l'Arduino, il peut communiquer avec l'ordinateur tant que la connexion est active (câble USB, ...)

Chapitre3

Présentation du projet

La dernière partie dans ce rapport, consiste à la description de la phase de mise en œuvre de l'application. Nous y illustrons les fonctionnalités de notre application à travers quelques interfaces.

Chapitre3 : Présentation du projet

Création de la base de données :

Pour marquer la présence des étudiants, on aura besoin d'une base de données contenant des informations relatives à chaque un. Elle contient des tableaux telle que (étudiants, Modules, Utilisateurs, Présence, Classes...). représente une capture de l'interface principale de PostgreSQL.

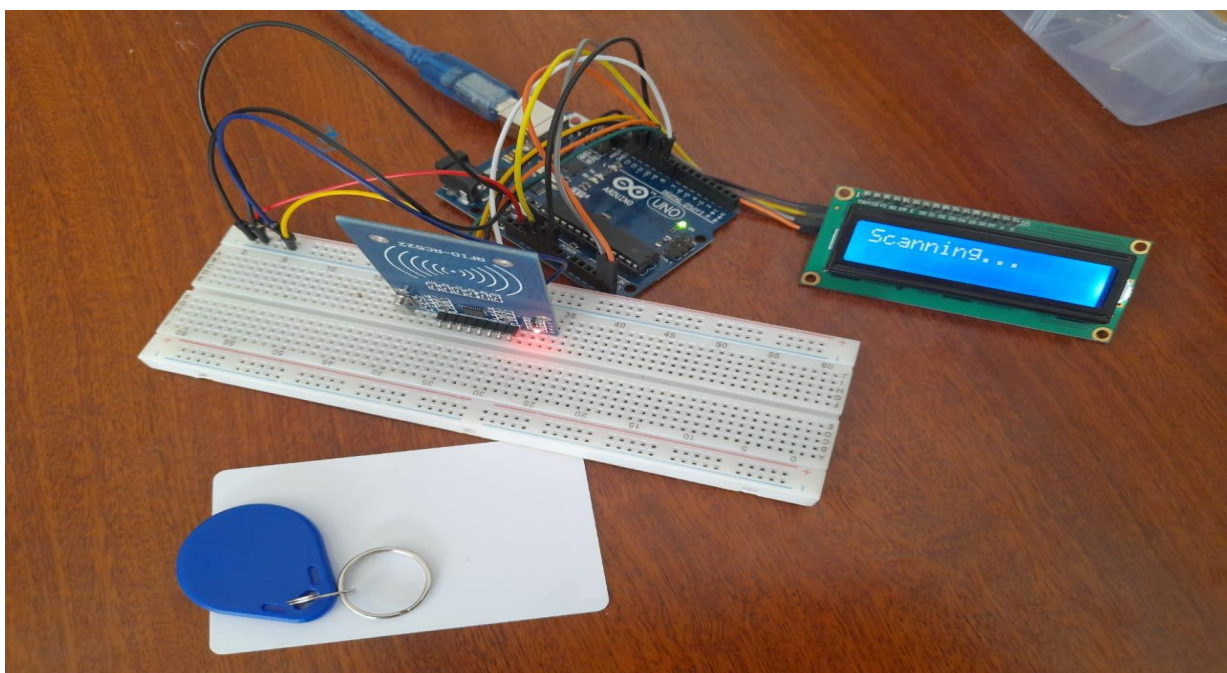
Prenons l'exemple tableau d'étudiant, qui peut contenir les informations personnelles telles que :

-Nom, email, iD, date de 1ère inscription, ...etc

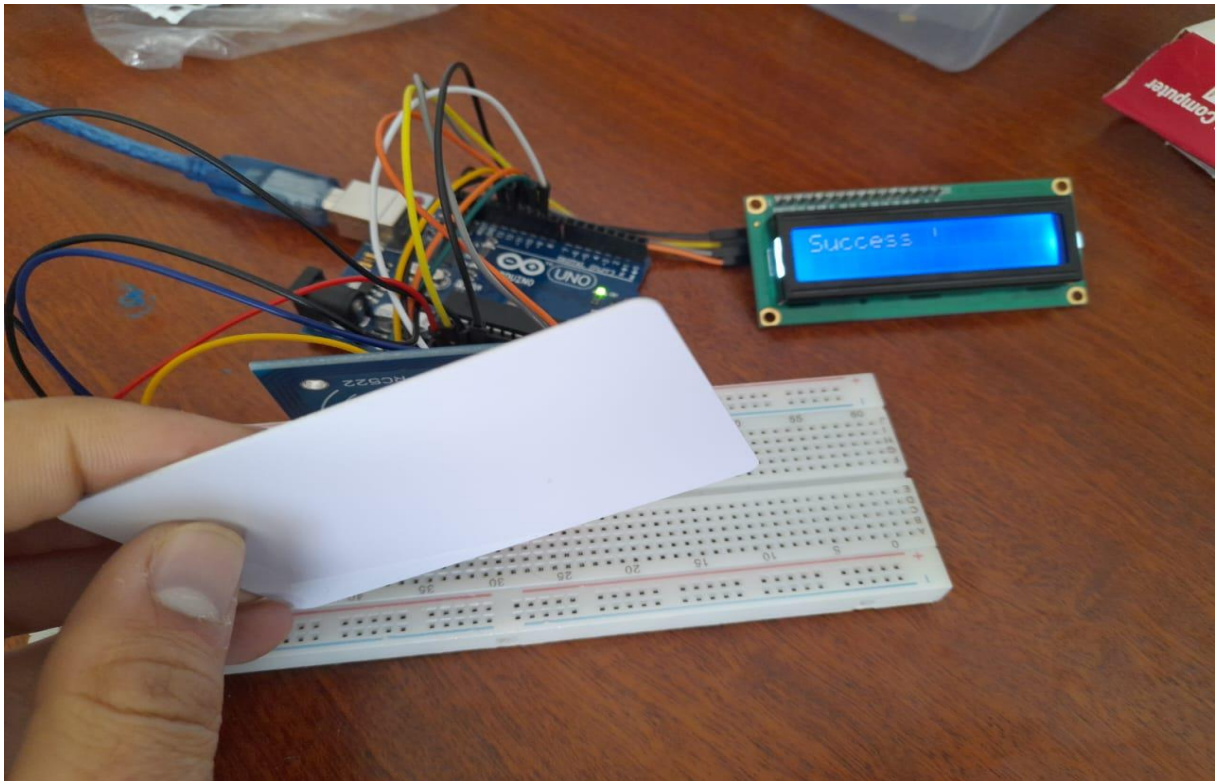
<<<	id big	created_at times	name text	email text	password text
	Filter	Filter	Filter	Filter	Filter

Programmation de microcontrôleur

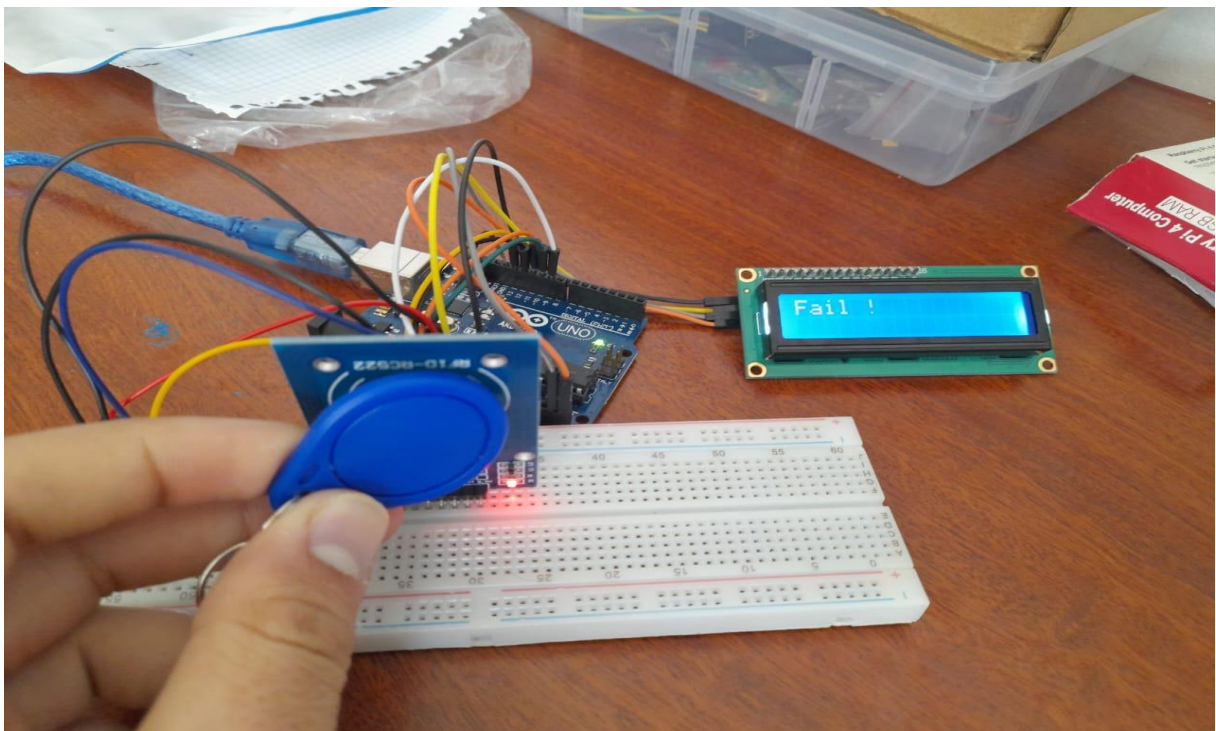
Avant d'entamer la partie programmation du microcontrôleur, nous passons au montage des composants du notre circuit dans une plaque d'essai (breadboard). L'illustre le branchement des composants (modules RFID, Wifi, LCD). Avec la carte Arduino UNO qui serve à la fois de le programmer via un câble USB (connecté au PC), et d'assurer son alimenter (5V).



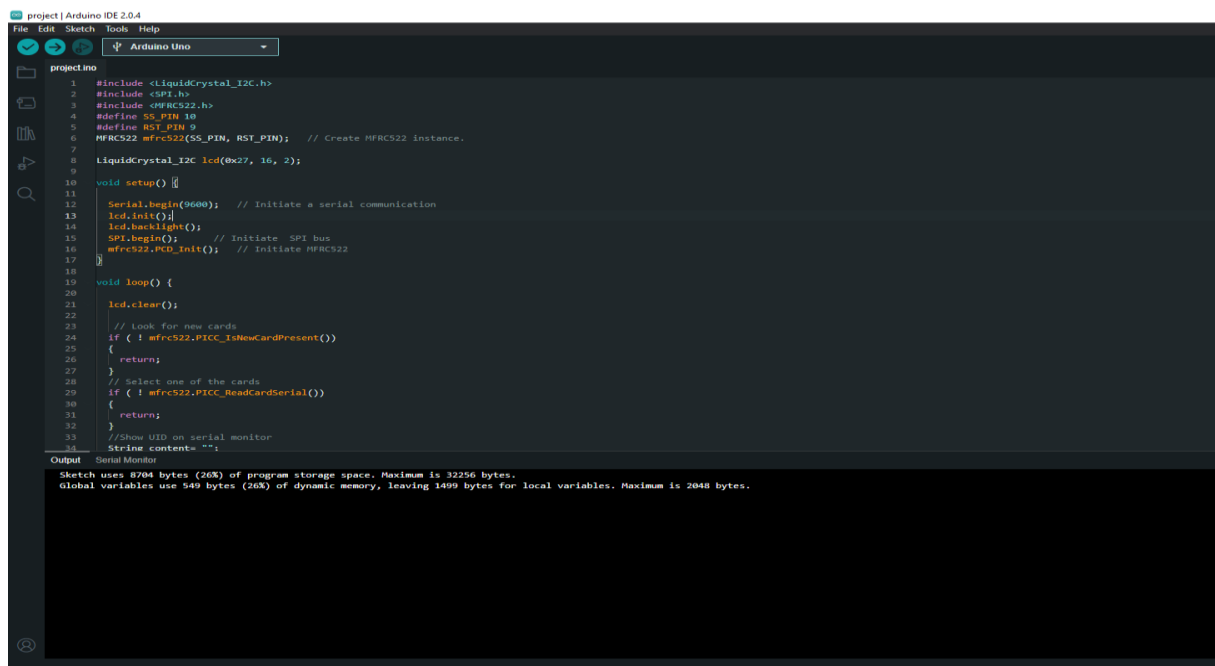
Une fois une carte pointe sa présence, il y a une vérification si ID existe dans la base de données et le LCD affiche Success :



Sinon LCD affiche Fail :



On exécute le code ci-dessus qui fait fonctionner notre simulation:



```
1 #include <LiquidCrystal_I2C.h>
2 #include <SPI.h>
3 #include <MFRC522.h>
4 #define SS_PIN 10
5 #define RST_PIN 9
6 MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
7
8 LiquidCrystal_I2C lcd(0x27, 16, 2);
9
10 void setup() {
11   Serial.begin(9600); // Initiate a serial communication
12   lcd.init();
13   lcd.backlight();
14   SPI.begin(); // Initiate SPI bus
15   mfrc522.PCD_Init(); // Initiate MFRC522
16 }
17
18 void loop() {
19   lcd.clear();
20
21   // Look for new cards
22   if (! mfrc522.PICC_IsNewCardPresent())
23   {
24     return;
25   }
26   // Select one of the cards
27   if (! mfrc522.PICC_ReadCardSerial())
28   {
29     return;
30   }
31   //Show UID on serial monitor
32   Serial.print("UID: ");
33   Serial.println(mfrc522.uid.toHex());
34 }
```

Sketch uses 8704 bytes (26%) of program storage space. Maximum is 32256 bytes.
Global variables use 549 bytes (26%) of dynamic memory, leaving 1499 bytes for local variables. Maximum is 2048 bytes.

Dans le montage précédent, nous avons rencontré plusieurs problèmes avec le module ESP8266 E01 ; qui reste toujours une solution pour d'autres systèmes vu son faible coût et simplicité de branchement et de configuration par rapport à d'autres versions plus récentes. Nous avons remarqué qu'après plusieurs tests, le module s'échauffe qui provoque à chaque fois un blocage (dysfonctionnement total du module) et qu'il faut une reprogrammation à nouveau.

Fonctionnement de système

Notre système de présence fonctionne d'une manière autonome une fois la carte d'étudiant est mise devant le lecteur RFID ce dernier détecte l'ID unique de l'étudiant, il le transmet directement vers la base de données via le module Wifi. Le système de présence est déployé à l'entrée de chaque classe, alimenté par la tension.

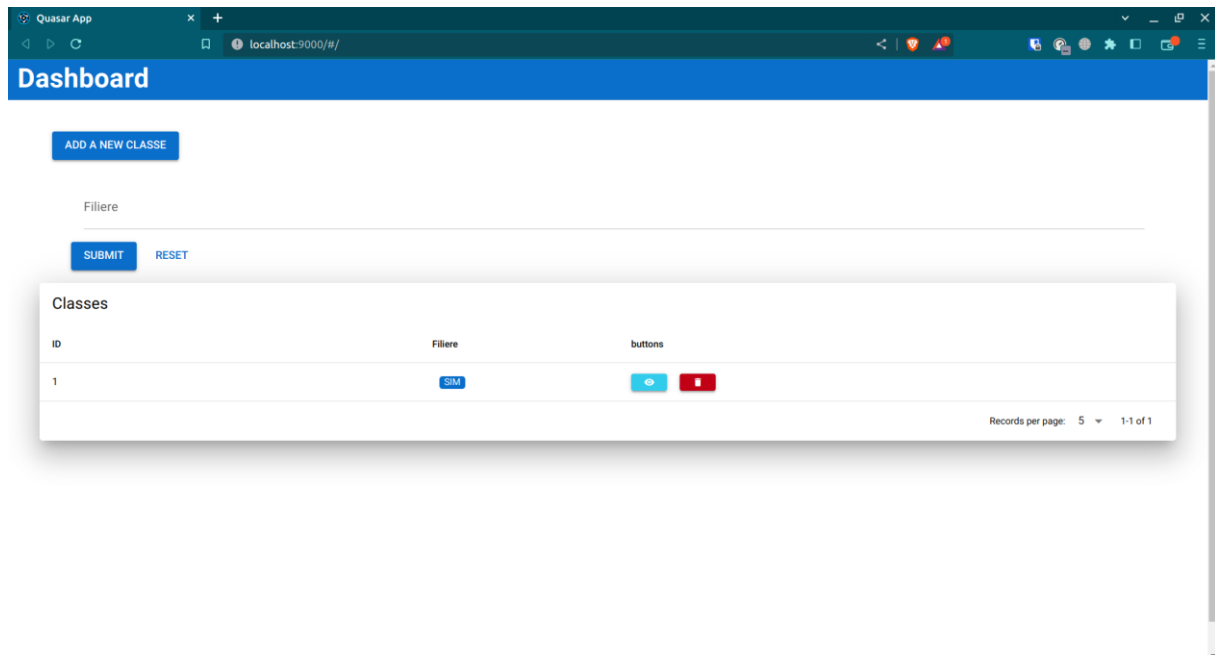
L'interface utilisateur est une application web multi plateforme, qui a besoin seulement d'un navigateur web pour la lancer, elle est simple et facile à utiliser depuis soit un PC de bureau, tablette ou bien un smartphone.

Notre application été hébergé dans Raspberry qui joue le rôle de serveur

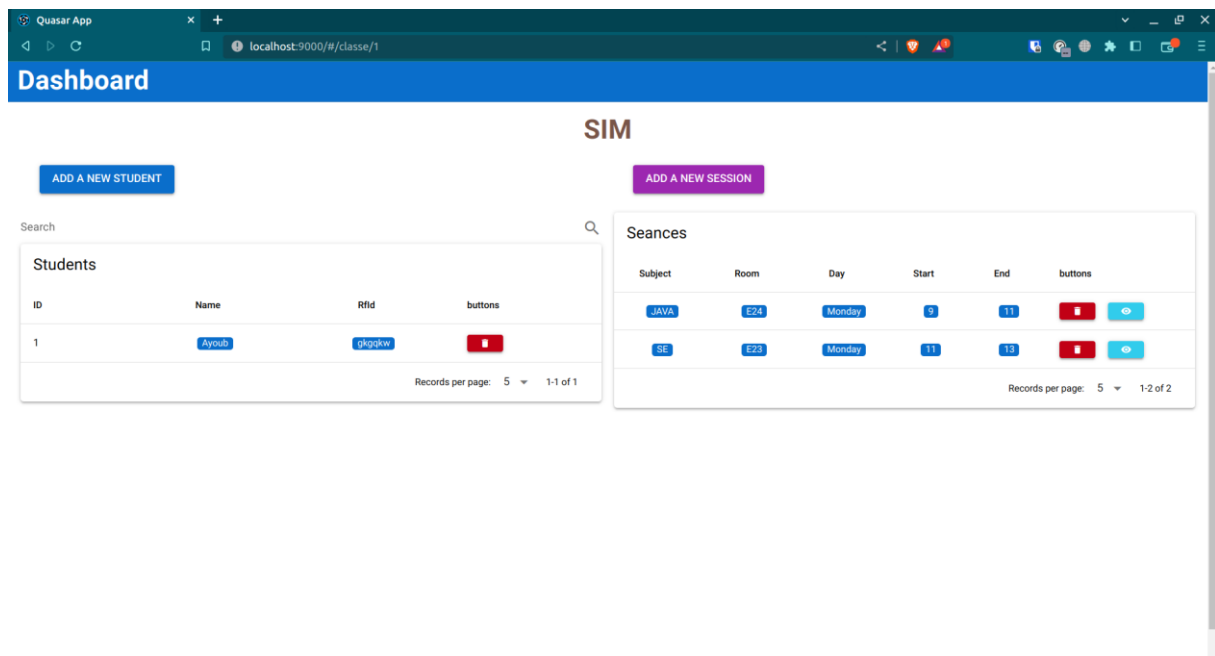


Les Interfaces :

L'interface principale qui permet de créer une nouvelle classe avec la possibilité de mettre à jour le filière et voir l'interface de étudiants et du séance avec le bouton détails



Interface de gestion des étudiants et de séance. Avec la possibilité de créer supprimer et modifier



Quasar App

localhost:9000/#/classe/1

Dashboard

SIM

ADD A NEW STUDENT

ADD A NEW SESSION

Search

Students

ID	Name	Rfid	buttons
1	Ayoub	gkqgkw	
2	Mouncef	H7GFDS4	

Records per page: 5 1-2 of 2

Seances

Subject	Room	Day	Start	End	buttons
JAVA	E24	Monday	9	11	
SE	E23	Monday	11	13	

Records per page: 5 1-2 of 2

Quasar App

localhost:9000/#/classe/1

Dashboard

SIM

ADD A NEW STUDENT

ADD A NEW SESSION

Search

Students

ID	Name	Rfid	buttons
1	Ayoub	gkqgkw	
2	Mouncef	Edaghgfa	

Records per page: 5 1-2 of 2

Matiere

Salle

Jour

Start Hour

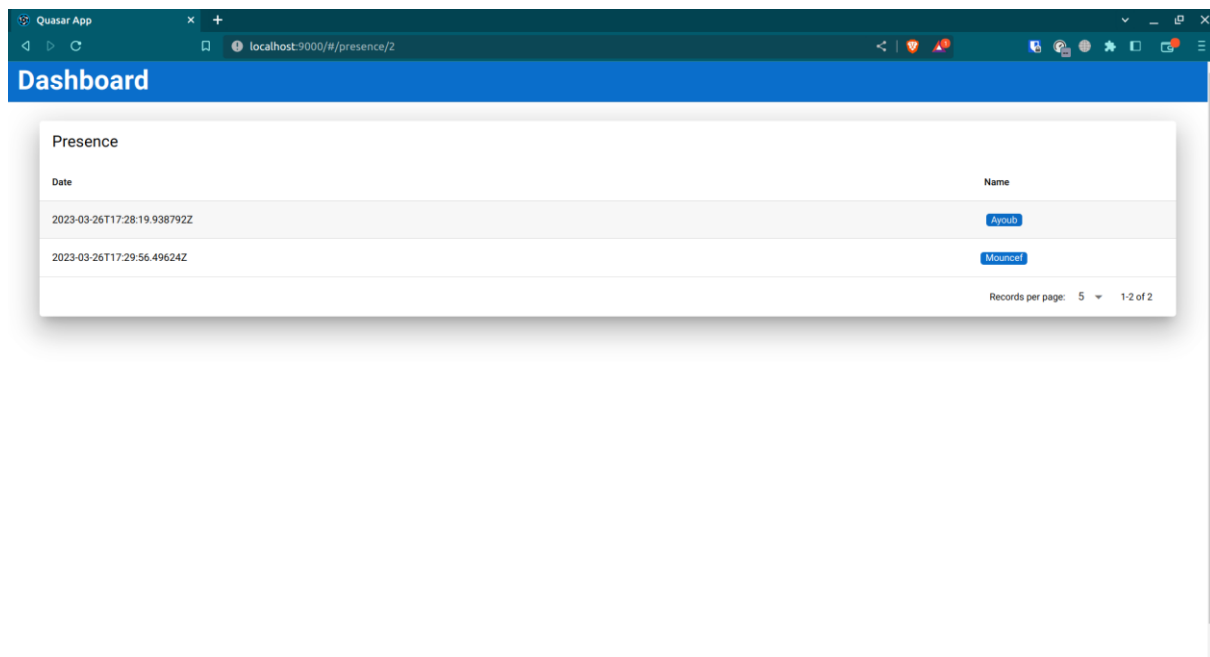
End Hour

SUBMIT RESET

Seances

Subject	Room	Day	Start	End	buttons
JAVA	E24	Monday	9	11	
SE	E23	Monday	11	13	

L'interface d'affichage de détails de présence



The screenshot displays a web application interface for viewing presence details. The browser window shows the URL `localhost:9000/#/presence/2`. The application has a blue header bar with the word "Dashboard". Below the header, there is a table titled "Presence". The table has two columns: "Date" and "Name". It contains two records. The first record shows the date "2023-03-26T17:28:19.938792Z" and the name "Ayoub". The second record shows the date "2023-03-26T17:29:56.49624Z" and the name "Mouncef". At the bottom right of the table, it indicates "Records per page: 5" and "1-2 of 2".

Date	Name
2023-03-26T17:28:19.938792Z	Ayoub
2023-03-26T17:29:56.49624Z	Mouncef