

CHARRAT Romain

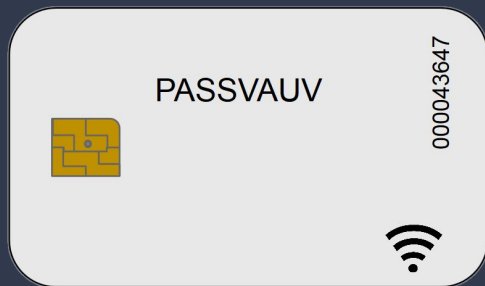
Terminale STI2D1

Spécialité : SIN

–Projet Carte de bus RFID–

avec SELLENET Lucas et
SAPINA Paul

Gestion d'une carte RFID et d'une BDD



-Sommaire-

2/19

- Introduction
 - Cahier des charges
 - Carte Mentale et diagramme des exigences
 - Diagramme IBD
 - Développement Durable
- Présentation matériel et logiciel
 - Arduino
 - Shield ethernet
 - MySQL
- Mise en pratique
 - Paramétrage du système
 - Création BDD
 - Programmes Arduino
 - Déclarations librairies
 - Init_DB
 - Query_r_trajet
 - Query_w_trajet
 - Programme de test
 - Exemple
- Conclusion

• Introduction

-Cahier des charges-

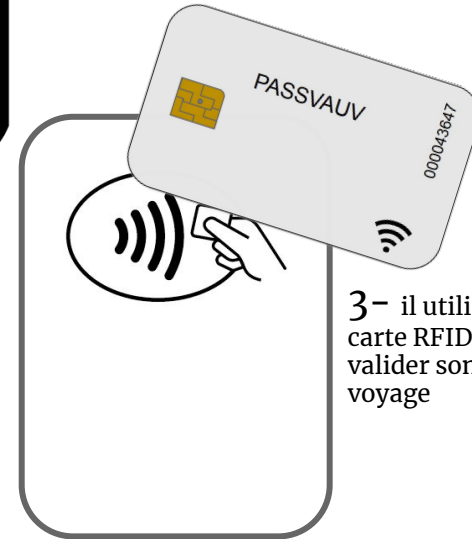


2- l'utilisateur veut monter dans le bus



1- Bus avec le conducteur

4- une requête est adressée à la BDD



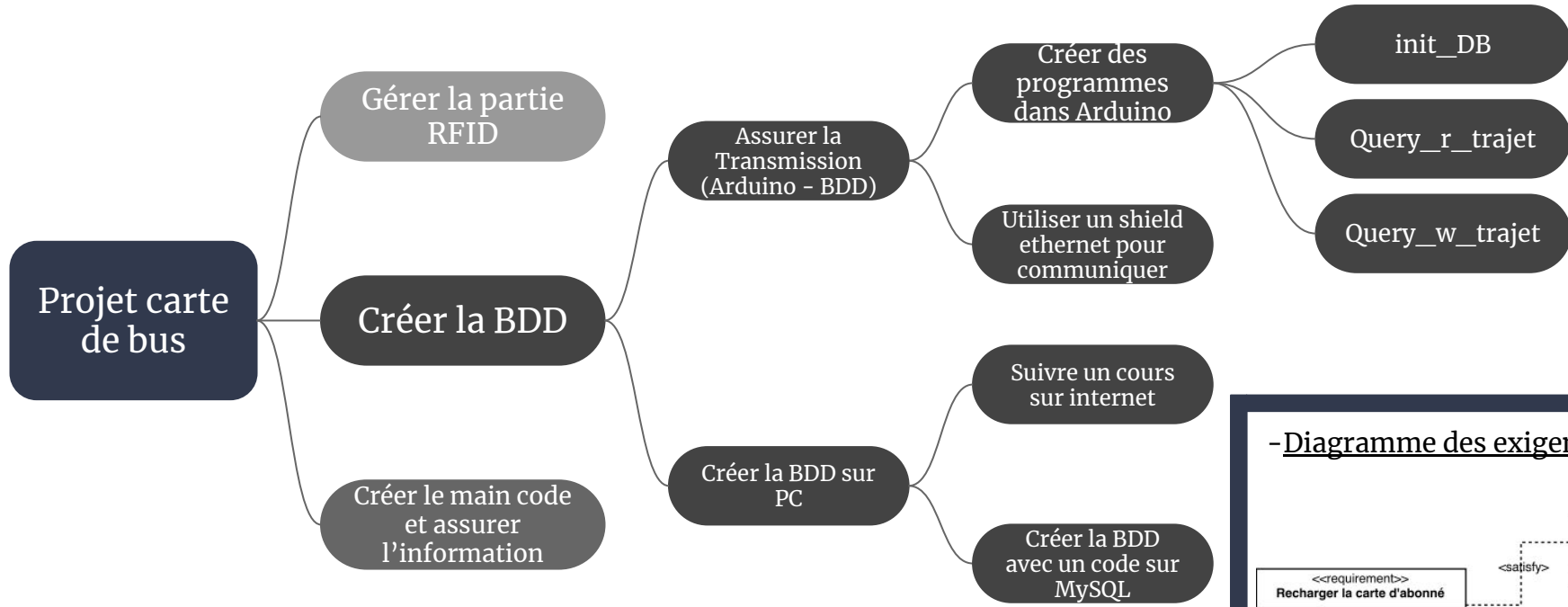
3- il utilise sa carte RFID pour valider son voyage

5- un buzzer et une LED informent le conducteur

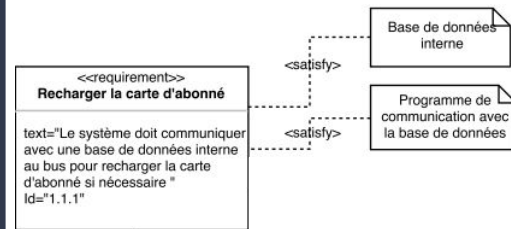


• Introduction

- Carte mentale et diagramme des exigences -



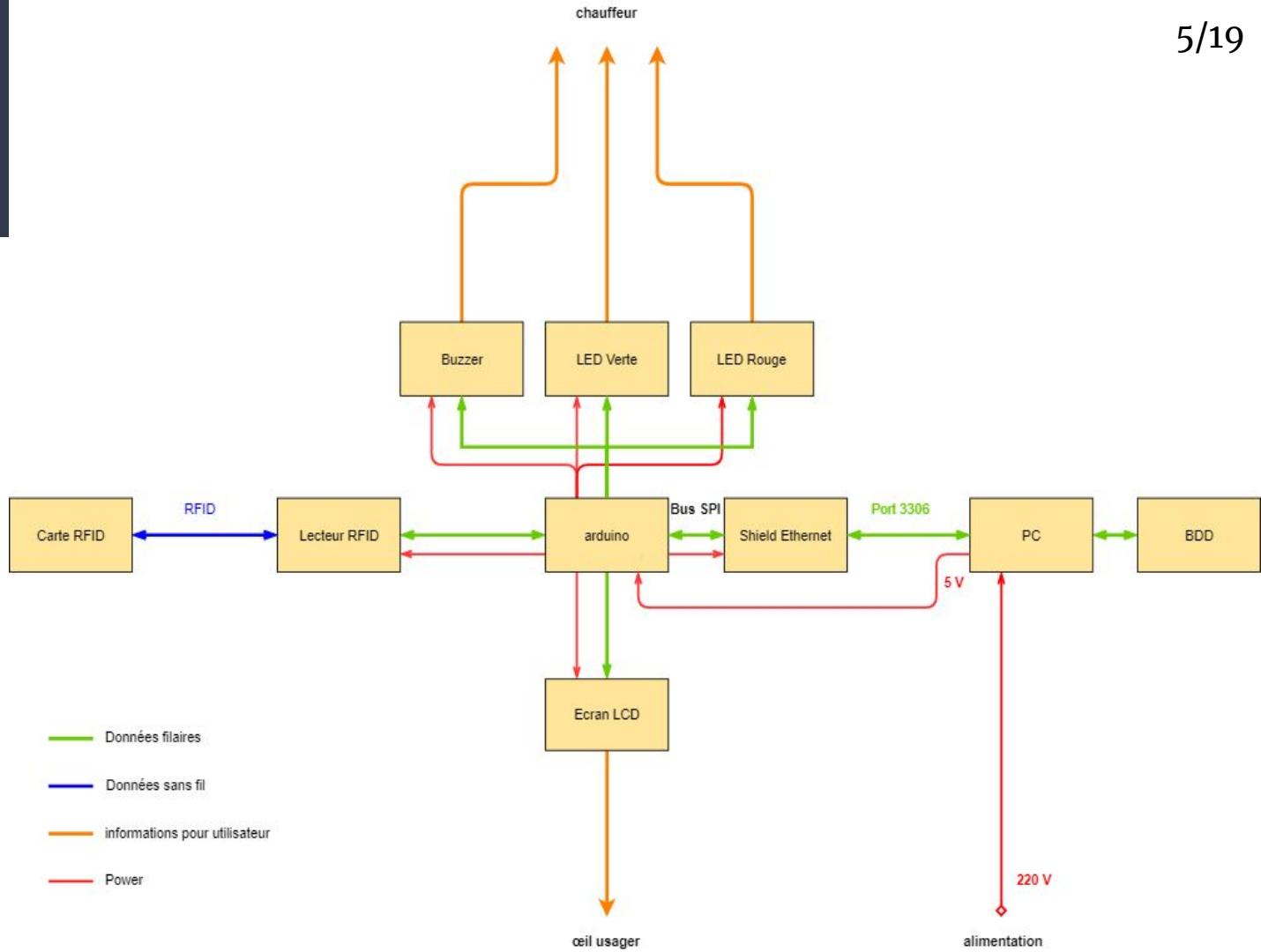
- Diagramme des exigences -



• Introduction

-Diagramme IBD-

5/19



• Introduction

-Développement Durable-



● Présentation matériel et logiciel

-Arduino-

- Carte électronique programmable -

- Des connecteurs entrées-sorties -

- Microcontrôleur fabriqué par Atmel -

- Port USB -



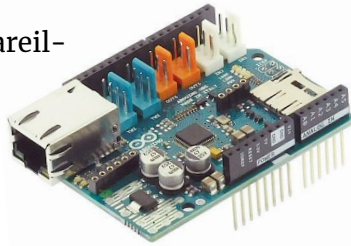
- Langage C++ -

- IDE permettant d'envoyer
des programmes sur la carte -

● Présentation matériel et logiciel

-Shield ethernet-

- Communication par ethernet avec un appareil-



- Communication Bus SPI -



- Pins 10,11,12,13 de l'arduino
utilisés (pin 4 pour carte SD) -

● Présentation matériel et logiciel

-MySQL-

- Système de gestion de base de données
relationnelles (SGBDR) -

- Langage SQL -



- Base



Table



Colonne



Donnée

● Mise en pratique

-Paramétrage du système-

- paramétrage IP carte ethernet du PC (serveur MySQL) -

Propriétés de : Protocole Internet version 4 (TCP/IPv4)

Général

Les paramètres IP peuvent être déterminés automatiquement si votre réseau le permet. Sinon, vous devez demander les paramètres IP appropriés à votre administrateur réseau.

☐ Obtenir une adresse IP automatiquement

☒ Utiliser l'adresse IP suivante :

Adresse IP : 192 , 168 , 0 , 10

Masque de sous-réseau : 255 , 255 , 255 , 0

Passerelle par défaut : . . .

☐ Obtenir les adresses des serveurs DNS automatiquement

☒ Utiliser l'adresse de serveur DNS suivante :

Serveur DNS préféré : . . .

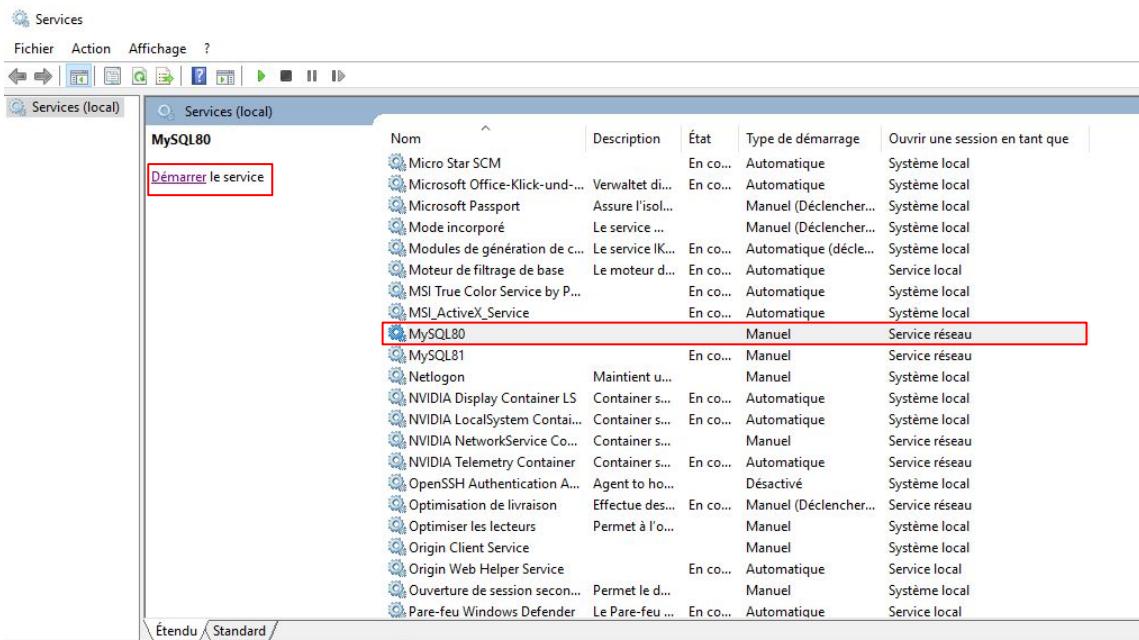
Serveur DNS auxiliaire : . . .

☐ Valider les paramètres en quittant

Avancé...

OK Annuler

- Démarrer service MySQL -



● Mise en pratique

-Création BDD-

-CREATION BASE-

→ CREATE DATABASE **passvauv** CHARACTER SET 'utf8';

-CREATION UTILISATEUR-

→ CREATE USER '**toto**'@'**192.168.0.143**' IDENTIFIED BY '**mot_de_passe**';
GRANT ALL PRIVILEGES ON **passvauv.*** TO '**toto**'@'**192.168.0.143**';

-AUTHENTICATION SHA1-

→ ALTER USER '**toto**'@'**192.168.0.143**' IDENTIFIED WITH mysql_native_password BY '**mot_de_passe**';
FLUSH PRIVILEGES;

-CREATION TABLE-

→ CREATE TABLE user (
id_card CHAR(8) NOT NULL,
nom VARCHAR(30) NOT NULL,
prenom VARCHAR(30),
trajet TINYINT UNSIGNED,
PRIMARY KEY (id_card)
)
ENGINE=INNODB;

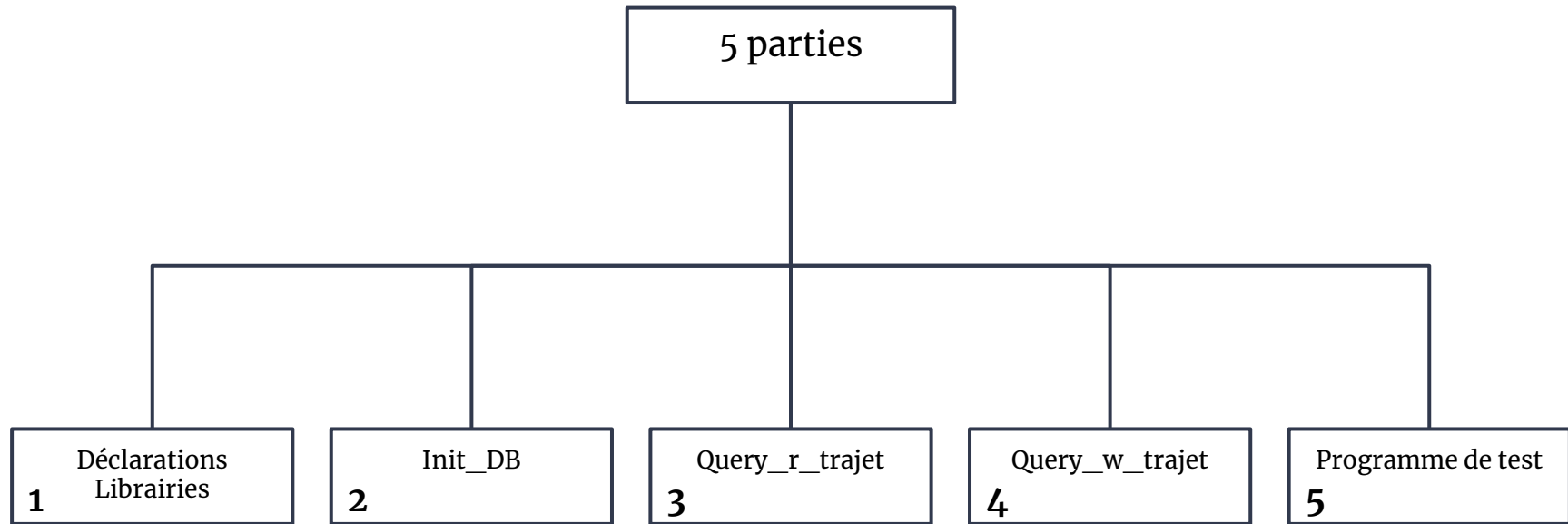
-CREATION DONNÉES-

→ INSERT INTO user (id_card, nom, prenom, trajet)
VALUES ('AABBCCDD', 'family', 'toto', '10'),
('11223344', 'famille', 'tata', '0');

-EXEMPLE-

id_card	nom	prenom	trajet
11223344	famille	tata	0
AABBCCDD	family	toto	10

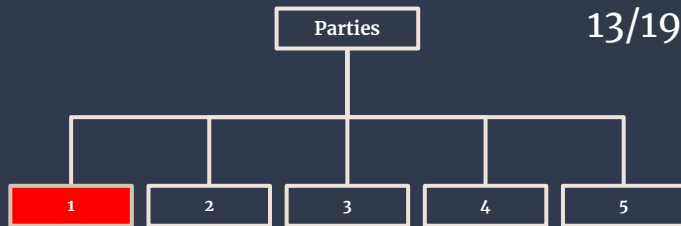
- Mise en pratique
 - Programmes Arduino -



● Mise en pratique

- Programmes Arduino -

1 → Déclarations Librairies



13/19

```
// Inclusion de la librairie du Shield Ethernet
```

```
#include <Ethernet.h>
```

```
// Inclusion de la librairie MySQL
```

```
#include <MySQL_Connection.h>
```

```
#include <MySQL_Cursor.h>
```

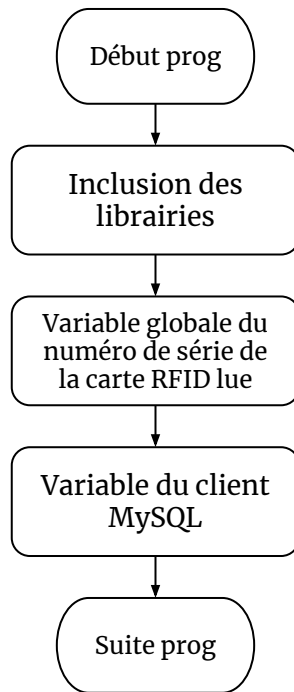
```
// Variable globale du numéro de série de la carte RFID lue
```

```
char id_card[] = "AABBCCDD";
```

```
// Variable du client MySQL
```

```
EthernetClient client;
```

```
MySQL_Connection conn((Client*)&client);
```

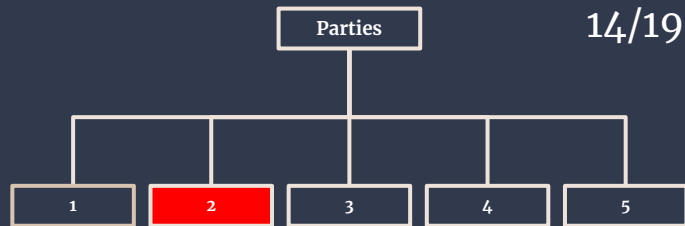


● Mise en pratique

- Programmes Arduino -

2 → Init_DB

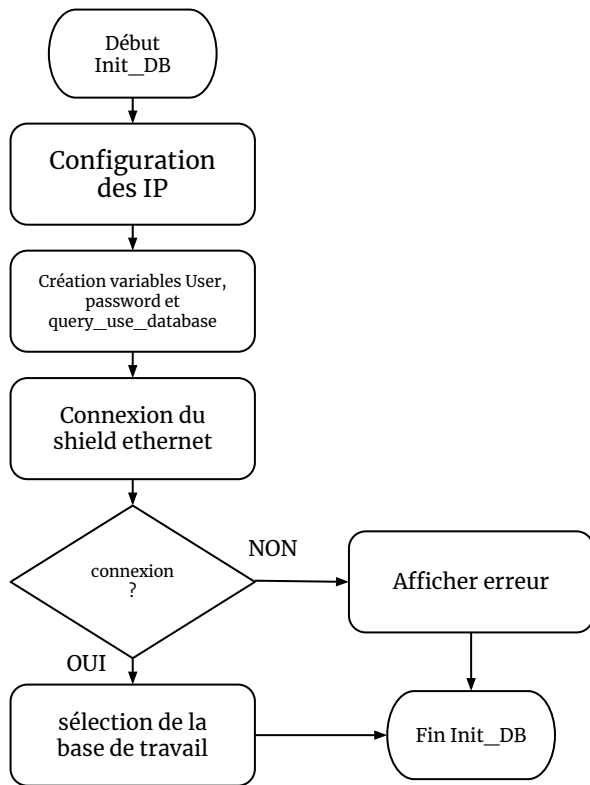
14/19



```
bool init_DB()
{
    // Adresse mac du shield ethernet
    byte mac_addr[] = { 0x90, 0xA2, 0xDA, 0x10, 0x57, 0x5D };

    IPAddress server_addr(192, 168, 0, 10);    // IP du serveur MySQL
    IPAddress ip(192, 168, 0, 143);            // Definition IP fixe du shield Ethernet
    char user[] = "toto";                      // MySQL user login username
    char password[] = "panda";                // MySQL user login password
    char query_use_database[] = "USE passvauv"; // Utilisation de la base "passvauv"

    // Connexion du shield Ethernet
    Ethernet.begin(mac_addr, ip);
    Serial.println("Connecting...");
    if (conn.connect(server_addr, 3306, user, password)) // Si la connexion a eu lieu
    {
        // Selection de la base de travail si la connexion DB a réussie
        MySQL_Cursor *cur = new MySQL_Cursor(&conn);
        cur->execute(query_use_database);
        delete cur; // Liberation memoire
        return true;
    }
    else
    {
        // Afficher erreur
        Serial.println("Connection failed.");
        return false;
    }
}
```

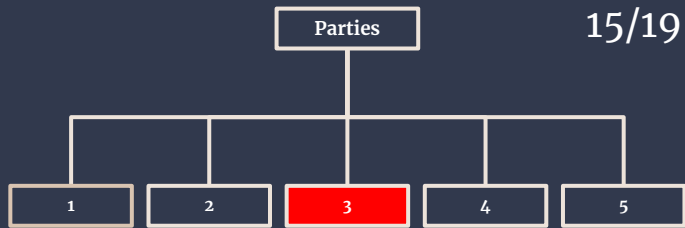


● Mise en pratique

- Programmes Arduino -

3 → Query_r_trajet

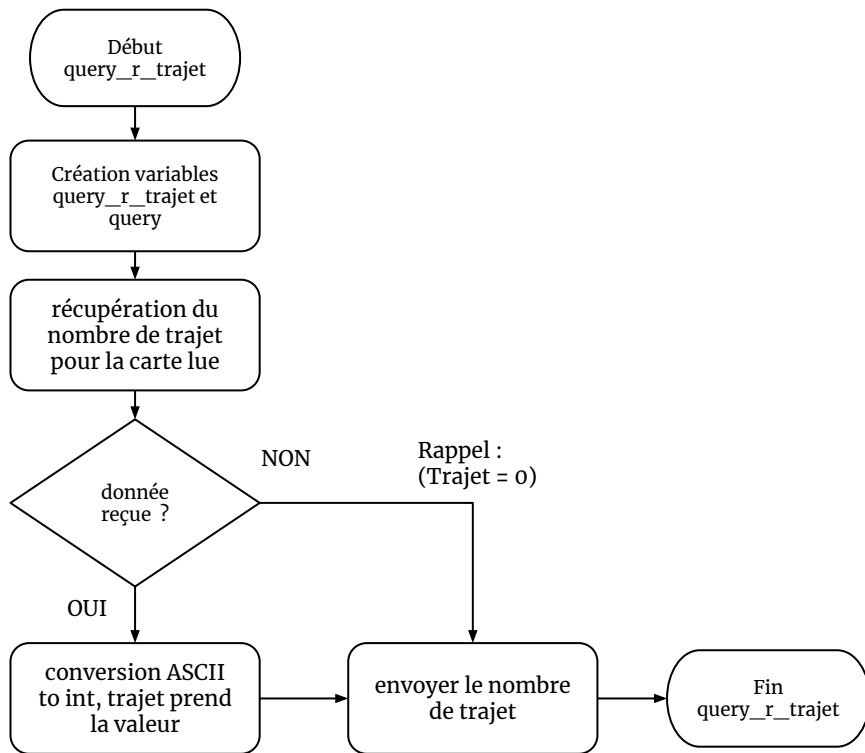
15/19



```
int query_r_trajet()
{
    // Creation variables
    char query_r_trajet[] = "SELECT trajet FROM passvauv.user WHERE id_card = '%s'";
    char query[75];

    // Recuperation du nombre de trajet pour la carte lue
    row_values *row = NULL;
    int trajet = 0; // Creation variable trajet
    MySQL_Cursor *cur = new MySQL_Cursor(&conn);
    sprintf(query, query_r_trajet, id_card); // Creation query
    if (cur->execute(query)) // Execution query
    {
        column_names *columns = cur->get_columns();
        row = cur->get_next_row();
        if (row != NULL)
        {
            trajet = atoi(row->values[0]); // Conversion ASCII to int
        }
    }
    delete cur; // Liberation memoire

    return trajet; // Envoyer le nombre de trajet
}
```

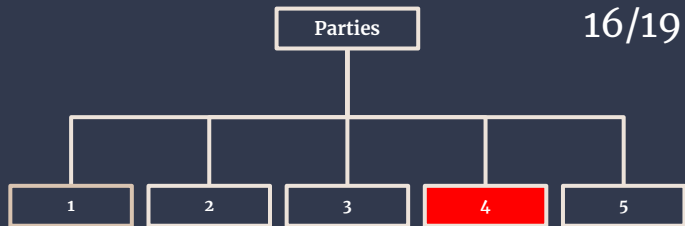


● Mise en pratique

- Programmes Arduino -

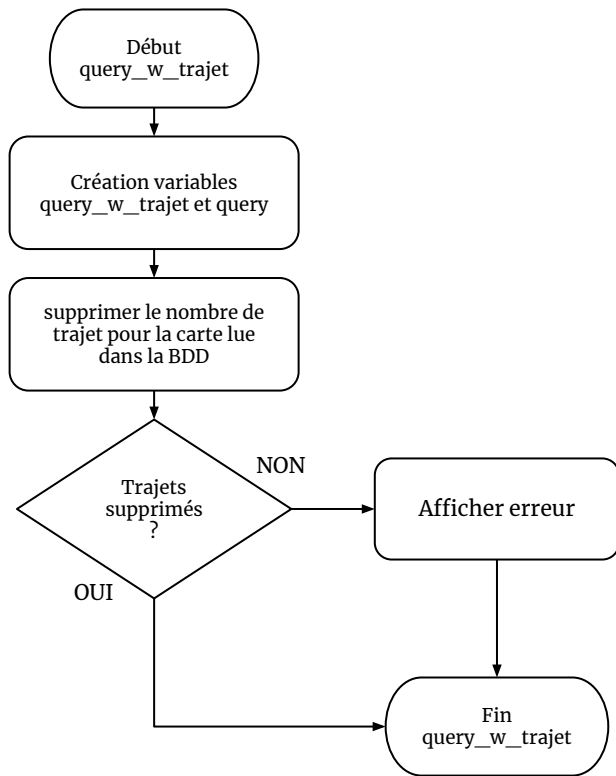
4 → Query_w_trajet

16/19



```
void query_w_trajet()
{
    // Creation variables
    char query_w_trajet[] = "UPDATE user SET trajet=0 WHERE id_card = '%s'";
    char query[75];

    // Supprimer le nombre de trajet pour la carte lue dans la BDD
    MySQL_Cursor *cur = new MySQL_Cursor(&conn);
    sprintf(query, query_w_trajet, id_card); // Creation query
    if (cur->execute(query) == false) // Execution Query
    {
        // Afficher erreur
        Serial.println("Erreur, impossible de supprimer");
    }
    delete cur; // Liberation memoire
}
```

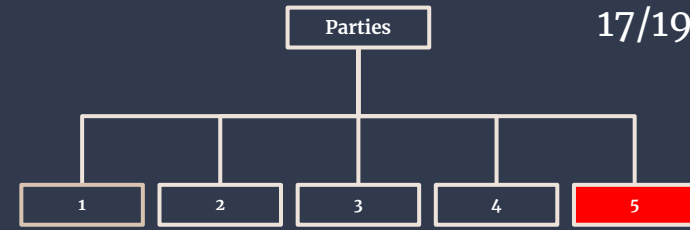


● Mise en pratique

- Programmes Arduino -

5 → Programme de test

17/19



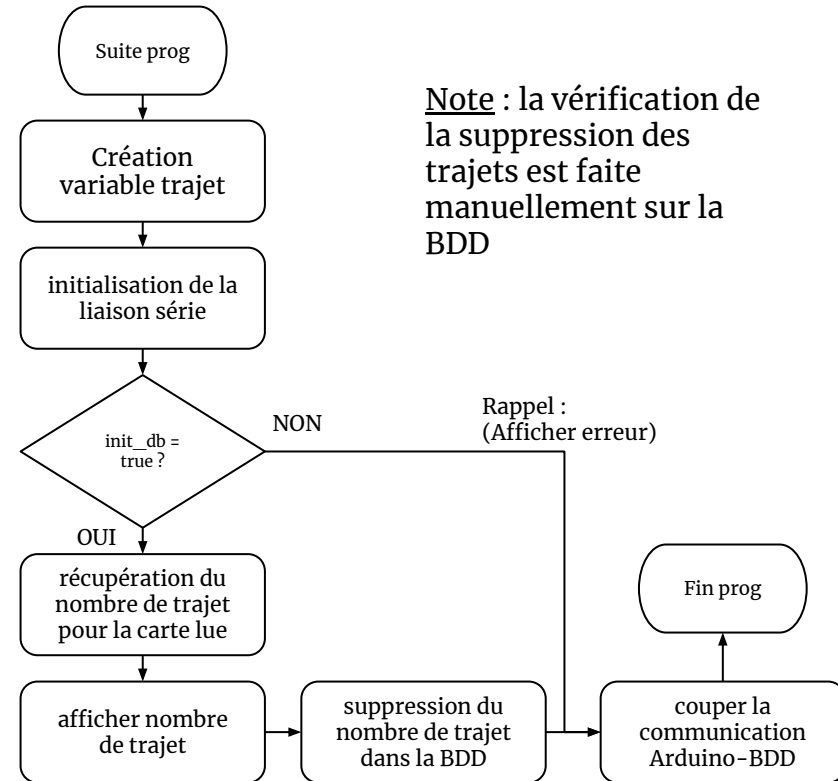
```
void setup()
{
  int trajet; //creation variable trajet

  // initialisation de la liaison serie
  Serial.begin(9600);
  while (!Serial); // Attendre que la liaison soit effectuée
  Serial.println("Serial open");

  if (init_DB()==true) // Si init_DB est effectué
  {
    Serial.println("*****");
    trajet = query_r_trajet(); // Trajet prend la valeur que query_r_trajet envoi

    // Afficher nombre de trajet
    Serial.print("nbre de trajet = ");
    Serial.println(trajet);

    query_w_trajet(); // Effectuer la suppression du nombre de trajet dans la BDD
  }
  conn.close(); // Couper la communication entre le Shield ethernet et la BDD
}
```



● Mise en pratique

-Exemple-

- Images du moniteur Série -

- Fonctionnement normal -

id_card	nom	prenom	trajet
11223344	famille	tata	0
AABBCCDD	family	toto	10

```
COM4

Serial open
Connecting...
Connected to server version 8.0.15
*****
nbre de trajet = 10
Disconnected.
```

☒ Défilement automatique ☐ Afficher l'horodatage

id_card	nom	prenom	trajet
11223344	famille	tata	0
AABBCCDD	family	toto	0

- Cas d'erreur -

id_card	nom	prenom	trajet
11223344	famille	tata	0
AABBCCDD	family	toto	10

```
COM4

Serial open
Connecting...
Connection failed.
```

☒ Défilement automatique ☐ Afficher l'horodatage

id_card	nom	prenom	trajet
11223344	famille	tata	0
AABBCCDD	family	toto	10

• Conclusion

19/19



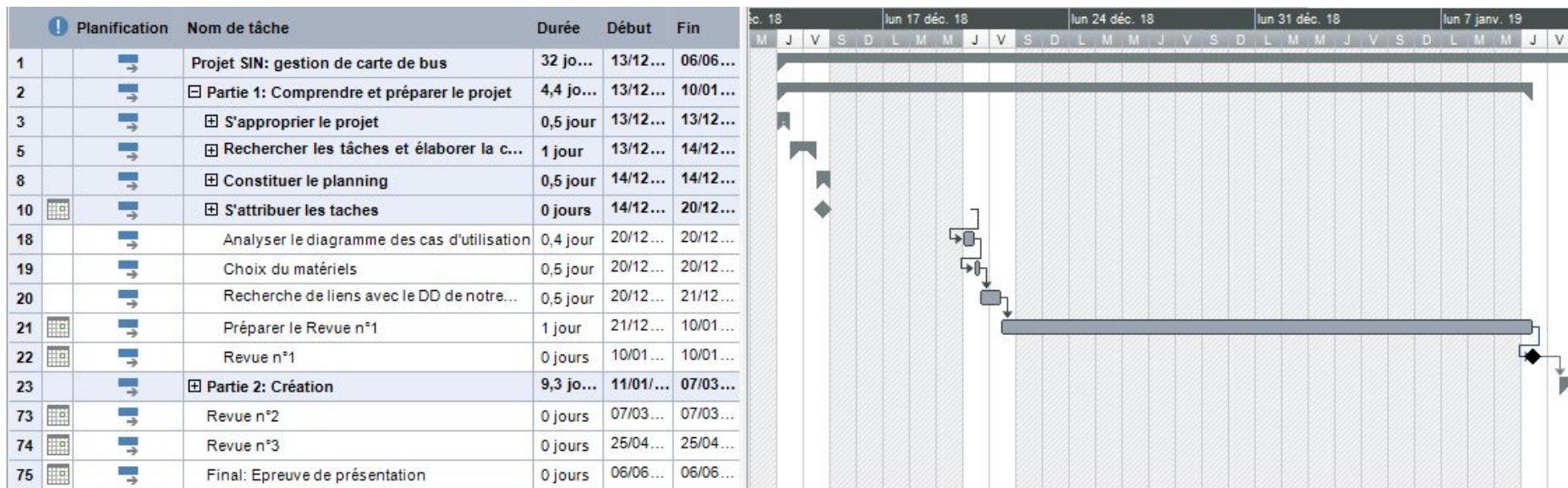
=



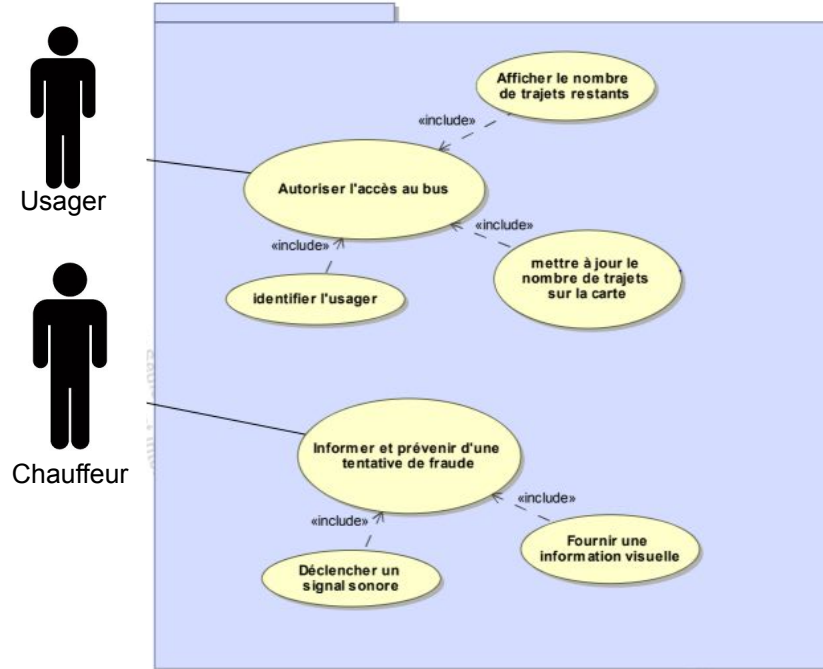
Merci de votre attention.

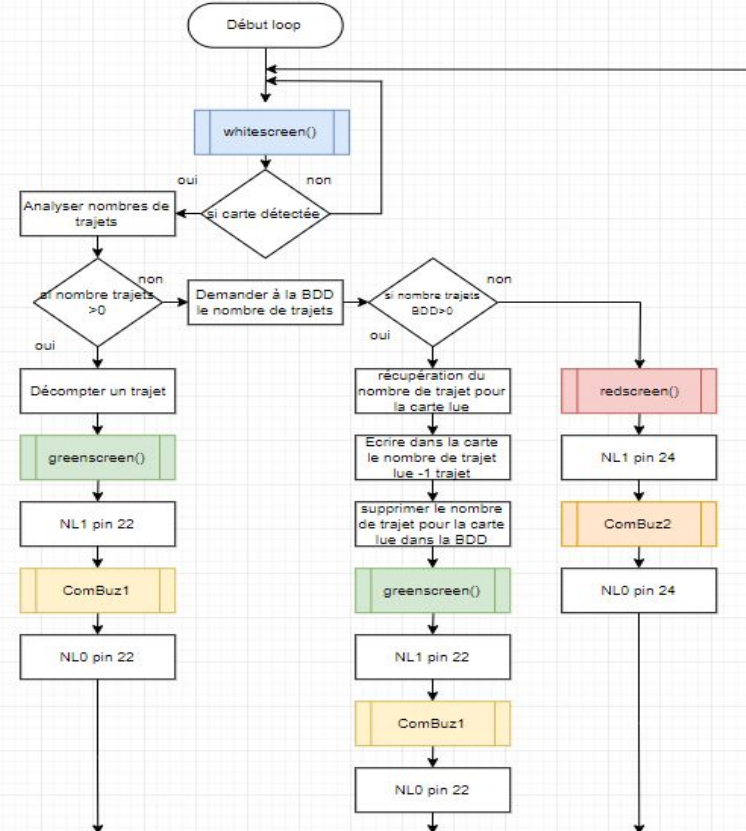
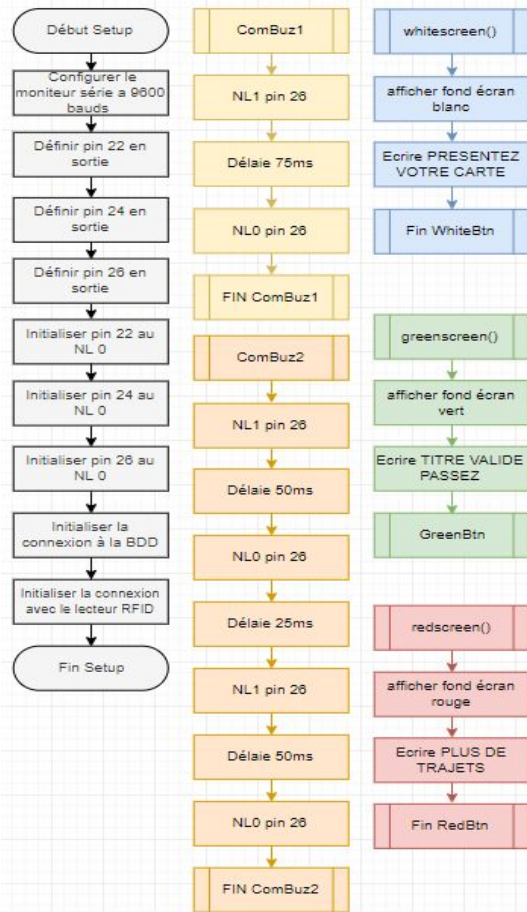
- ANNEXES -

● Gantt



- Diagramme des cas d'utilisation





● Organigramme du projet