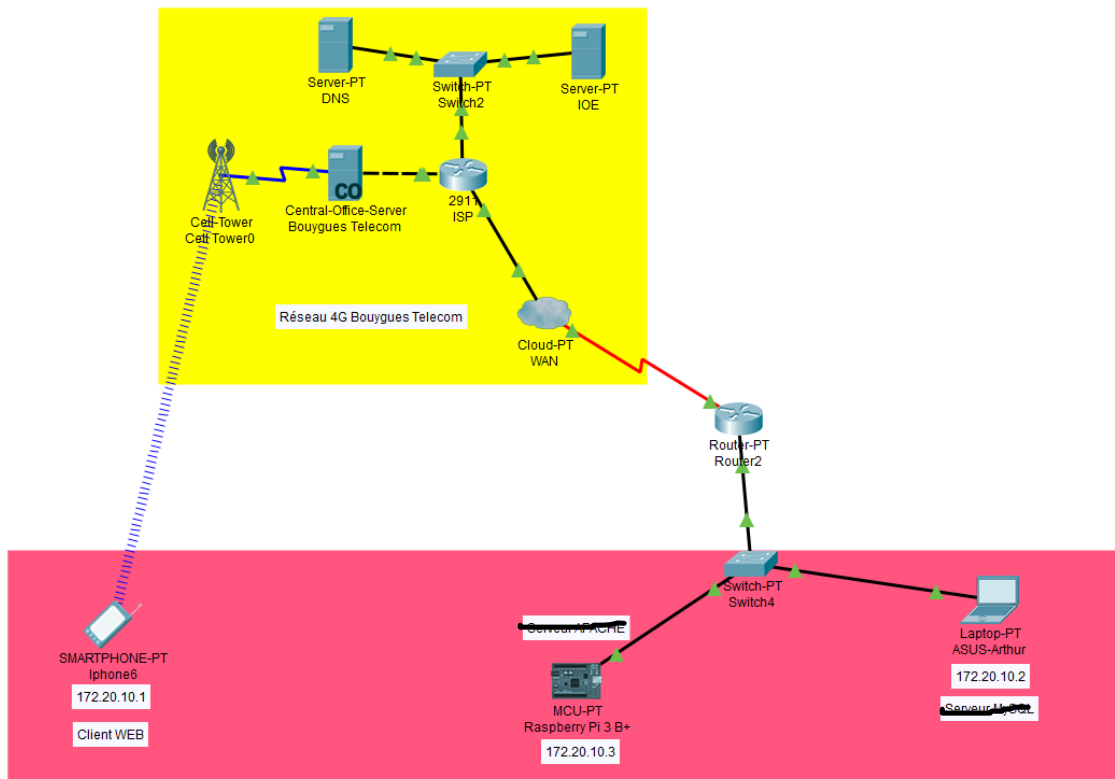


# Préparation à l'utilisation du contexte GSB

Schéma de l'architecture réseau :



Description des trois machines :

- 1) Serveur MariaDB-MySQL sous Raspbian avec proftpd (172.20.10.3)
- 2) Serveur WEB Apache 2.4 sous W10 avec FileZilla (172.20.10.2)
- 3) Client WEB sous iOS avec Safari (172.20.10.1)

Description des deux réseaux :

- 1) Réseau **JAUNE** : Réseau interne à Bouygues Telecom me permettant d'obtenir une connexion 4G avec mon forfait de téléphone mobile.
- 2) Réseau **ROSE** : Réseau interne obtenu avec un partage de connexion 4G.

Particularité du réseau :

- Totalement déconnecté de l'infrastructure réseau du lycée = **Travail à la maison possible**
- Fonctionne localement **SANS CABLE RJ45** = **Moderne**
- Rapidité d'installation et de configuration = **Plus efficace à l'avenir**
- Accès à Internet

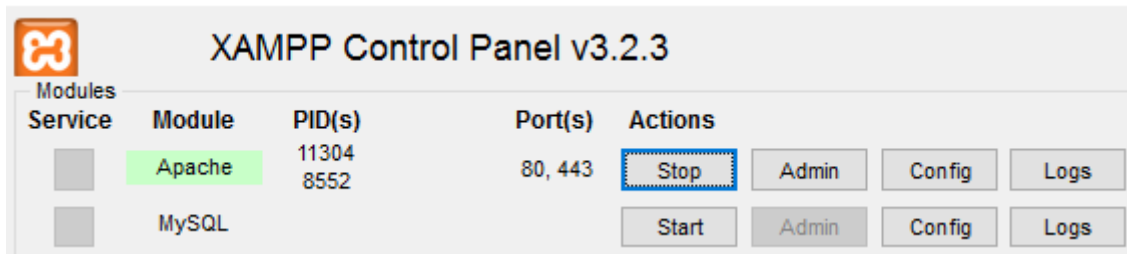
## SOMMAIRE

- I. Installation d'apache sous Windows 10
- II. Installation du serveur MySQL sur mon Raspberry Pi
- III. Installation du serveur FTP proftpd sous Raspbian
- IV. Installation du client FileZilla sur mon PC
- V. Installation du client SSH PuTTY et SSH sur Raspbian
- VI. Configuration du serveur MySQL
- VII. Configuration du serveur apache
- VIII. Test d'accès via le client web

## I. Installation du serveur apache sous Windows 10

Étant donné que XAMPP est installé sur mon PC, Apache 2.4 est installé avec la configuration de base

Pour installer XAMPP, veuillez cliquer sur : <https://www.apachefriends.org/xampp-files/7.3.9/xampp-windows-x64-7.3.9-0-VC15-installer.exe>



On vérifie que apache fonctionne bien :



### Index of /

	<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
	<a href="#">COC_dashboard/</a>	2019-06-13 21:11	-	
	<a href="#">ServeurPerso/</a>	2019-09-16 19:05	-	
	<a href="#">appliFrais/</a>	2019-09-03 10:58	-	
	<a href="#">concoursNumerique - ..&gt;</a>	2019-06-26 09:23	-	
	<a href="#">concoursNumerique/</a>	2019-06-25 11:06	-	
	<a href="#">portfolio/</a>	2019-09-18 09:47	-	
	<a href="#">stage1/</a>	2019-06-25 10:02	-	
	<a href="#">test_api/</a>	2019-07-21 18:02	-	

*Apache/2.4.39 (Win64) OpenSSL/1.1.1b PHP/7.3.5 Server at localhost Port 80*

## II. Installation du serveur MySQL sur mon Raspberry Pi

Mon Raspberry Pi fonctionne avec la distribution Raspbian, qui est une distribution inspirée de Debian. Il n'y a donc pas de paquet « mysql-server » ou « mysql ». Cependant, il existe un paquet « mariadb-server » qui utilise mysql.

```
pi@raspberrypi:~$ sudo apt install -y mariadb
pi@raspberrypi:~$ sudo mysql_secure_installation
```

On teste que le serveur fonctionne bien et mysql aussi :

```
pi@raspberrypi:~$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.1.38-MariaDB-0+deb9u1 Raspbian 9.0

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

pi@raspberrypi:~$ sudo service mariadb status
• mariadb.service - MariaDB 10.1.38 database server
  Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2019-09-20 11:01:31 CEST; 3min 7s ago
```

## III. Installation du serveur FTP proftpd sous Raspbian

```
pi@raspberrypi:~$ sudo apt install -y proftpd
```

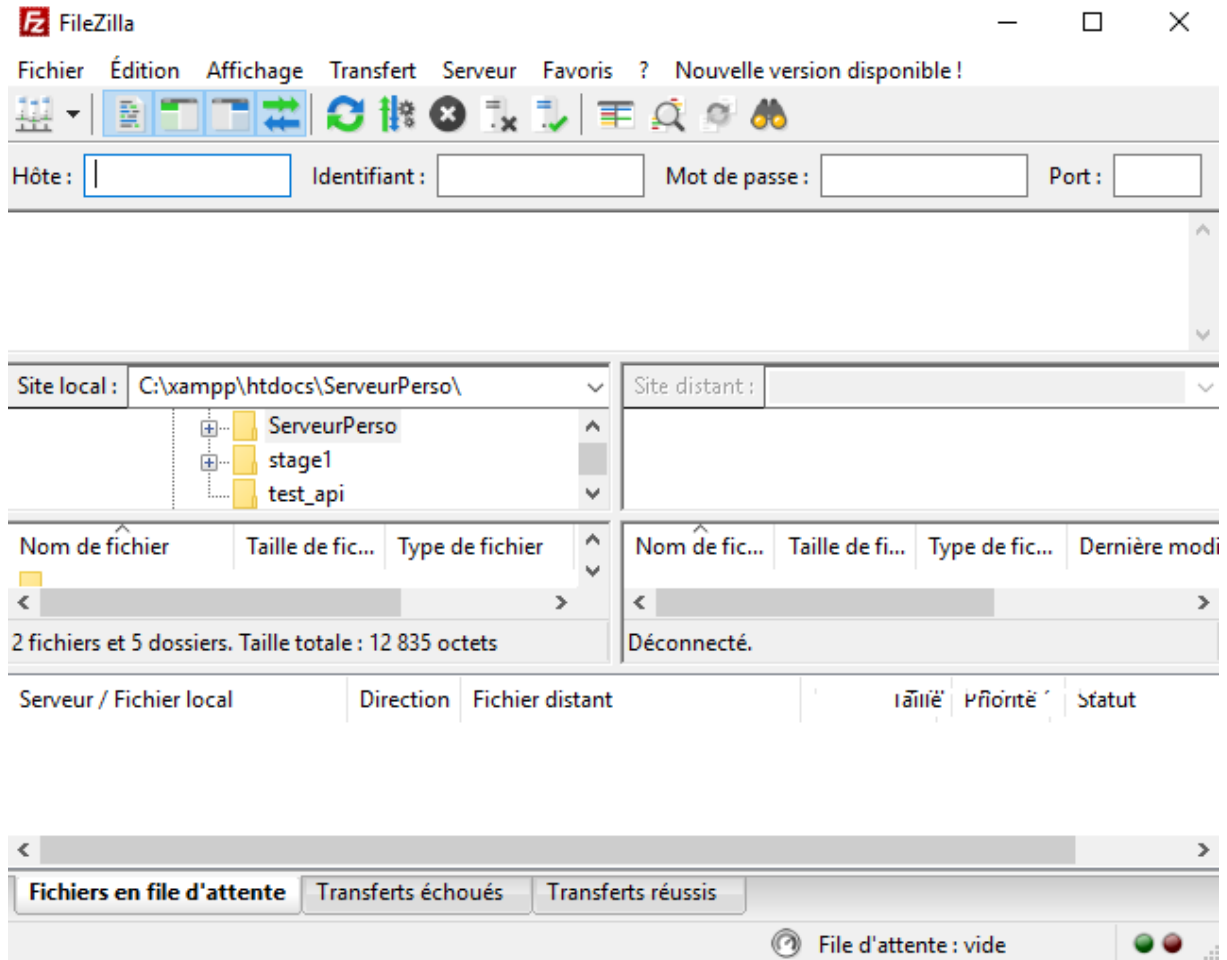
On vérifie que proftpd fonctionne bien :

```
pi@raspberrypi:~$ sudo service proftpd status
• proftpd.service - LSB: Starts ProFTPD daemon
  Loaded: loaded (/etc/init.d/proftpd; generated; vendor preset: enabled)
  Active: active (running) since Wed 2019-09-18 14:25:06 CEST; 21min ago
```

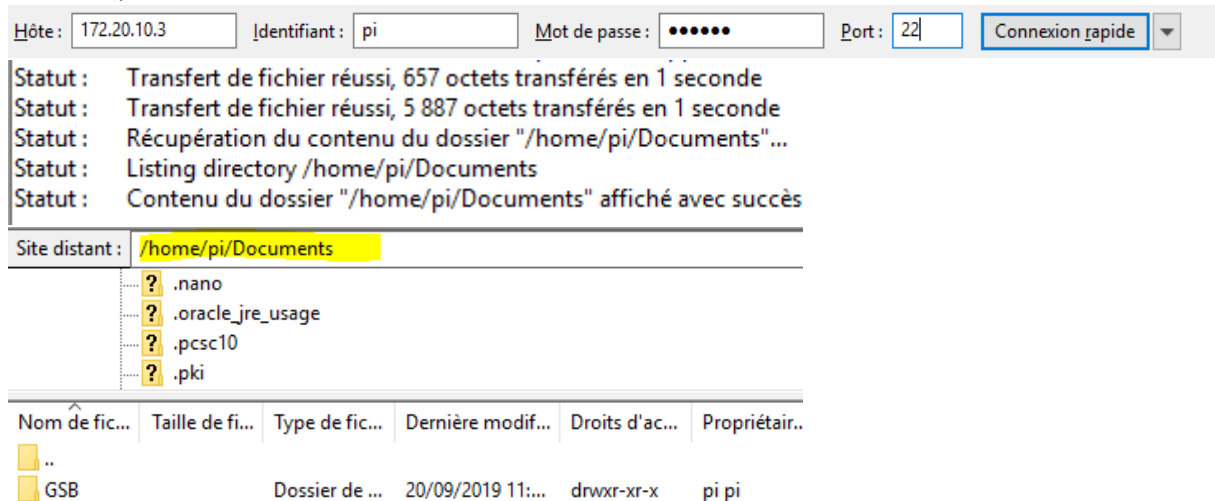
## IV. Installation du client FileZilla sur mon PC personnel

Pour télécharger FileZilla, veuillez cliquer ici : [https://download.filezilla-project.org/client/FileZilla\\_3.44.2\\_win64\\_sponsored-setup.exe](https://download.filezilla-project.org/client/FileZilla_3.44.2_win64_sponsored-setup.exe)

FileZilla Client nécessite aucunes configurations autres que l'hôte à joindre, le logiciel marche dès la fin du téléchargement :



On envoie ensuite le dossier comprenant les scripts de la future base de données en se connectant en SFTP (port 22 comme SSH)

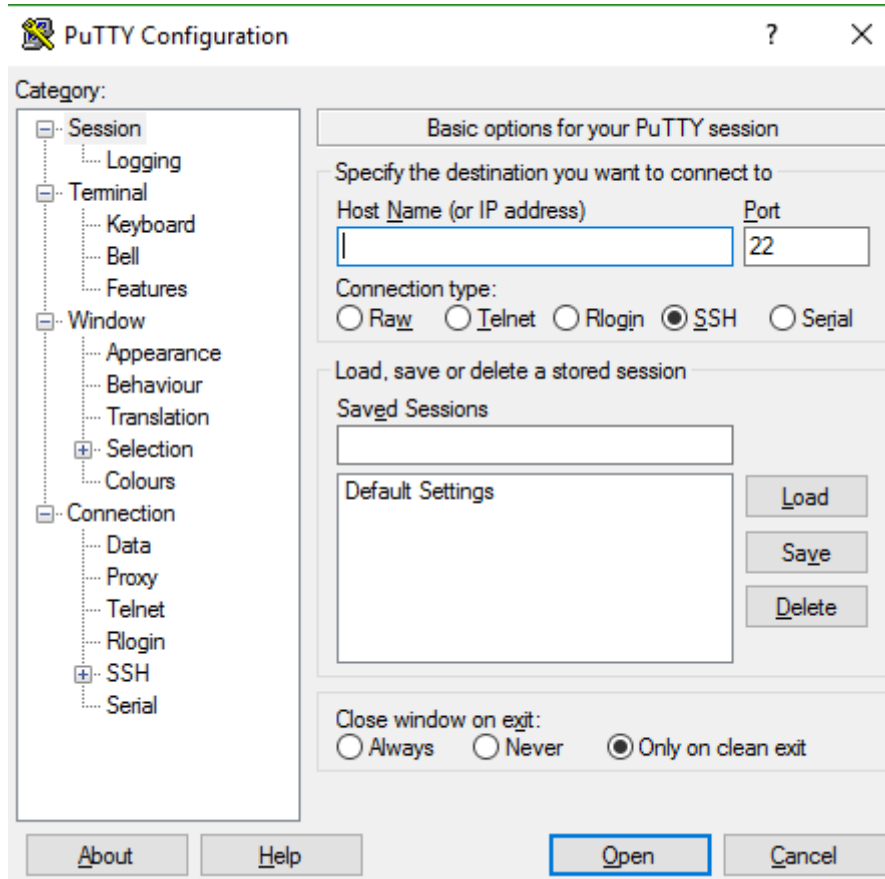


## V. Installation de PuTTY sur mon PC et SSH sur Raspbian

Pour installer PuTTY sur Windows, cliquez ici :

<https://the.earth.li/~sgtatham/putty/latest/w64/putty-64bit-0.72-installer.msi>

On vérifie que le logiciel fonctionne bien :



SSH est déjà installer sur Raspbian de base, mais la commande à réaliser est celle-ci :

```
pi@raspberrypi:~ $ sudo apt install ssh
```

On vérifie que ssh fonctionne bien :

```
pi@raspberrypi:~ $ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2019-09-17 23:17:06 CEST; 15h ago
```

Le protocole SSH nous permettra de communiquer entre les deux serveurs afin d'avoir un écran à disposition pour le Raspberry Pi et de pouvoir envoyer les scripts SQL vers le Raspberry pi en utilisant le protocole SFTP.

## VI. Configuration du serveur MySQL

On se connecte et on importe les scripts du contexte gsb :

```
MariaDB [(none)]> CREATE DATABASE 'bdGsb';
```

```
MariaDB [(none)]> source /home/pi/Documents/GSB/PPEApplicationFrais/Ressources/GSB-BDD/gsb_frais_structure.sql;
```

```
MariaDB [(none)]> source /home/pi/Documents/GSB/PPEApplicationFrais/Ressources/GSB-BDD/gsb_frais_insert_tables_statiques.sql;
```

On créer un utilisateur propre pour cette base de données :

```
MariaDB [bdGsb]> CREATE USER 'userGsb'@'localhost';  
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [bdGsb]> GRANT ALL PRIVILEGES ON bdGsb.* TO 'userGsb'@'localhost' IDENTIFIED BY 'secret';  
Query OK, 0 rows affected (0.00 sec)
```

Ainsi qu'un autre utilisateur accessible depuis le client :

```
MariaDB [bdGsb]> CREATE USER 'userGsb'@'172.20.10.1';  
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [bdGsb]> GRANT ALL PRIVILEGES ON bdGsb.* TO 'userGsb'@'172.20.10.1' IDENTIFIED BY 'secret';  
Query OK, 0 rows affected (0.00 sec)
```

Ensuite, on commente cette ligne pour que apache situé en 172.20.10.2 puisse se connecter avec la base de données :

```
pi@raspberrypi:/etc/mysql/mariadb.conf.d $ cd /etc/mysql/mariadb.conf.d/  
pi@raspberrypi:/etc/mysql/mariadb.conf.d $ sudo nano 50-server.cnf
```

```
# bind-address            = 127.0.0.1
```

## VII. Configuration du serveur apache

On commence par changer le script php de connexion à la base de données afin de l'adapter à notre architecture réseau :

```
function connecterServeurBD() {  
    $hote = "172.20.10.3"; // On ajoute l'ip du Raspberry où se trouve le serveur MariaDB  
    $login = "userGsb";  
    $mdp = "secret";  
    return mysqli_connect($hote, $login, $mdp);  
}
```

On place le dossier appliFrais dans : C:/xampp/htdocs/ et le serveur apache2.4 est opérationnel

## VIII. Test d'accès via le client web

Avec le client web Safari, nous nous rendons sur <http://172.20.10.2/appliFrais/cAccueil.php> et nous entrons par exemple le login **cbedos** et le mot de passe **gmhxd** pour le visiteur **Christian Bedos**

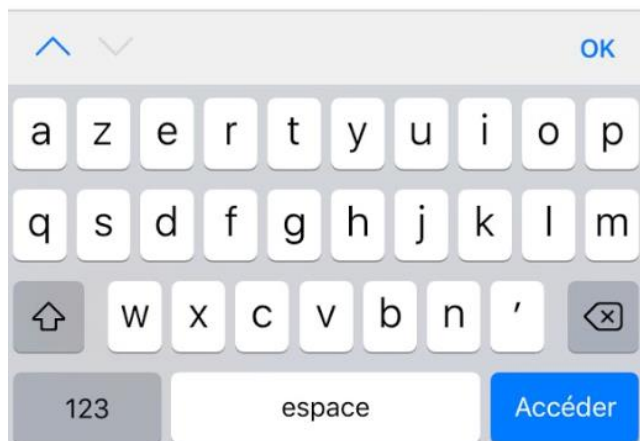
172.20.10.2

ication utilisateur

Login : cbedos

Mot de passe : ●●●●●

Valider



La connexion a réussi et notre architecture trois tiers est désormais fonctionnelle !