INSTRUCTION MANUAL

APPLICATIONS REQUIRED

- Arduino IDE
- Node.js
- Xampp

ARDUINO LIBRARIES REQUIRED

- Adafruit SSD1306
- Adafruit Fingerprint Sensor Library
- SimpleTimer by jfurcot

PROJECT FILES REQUIRED

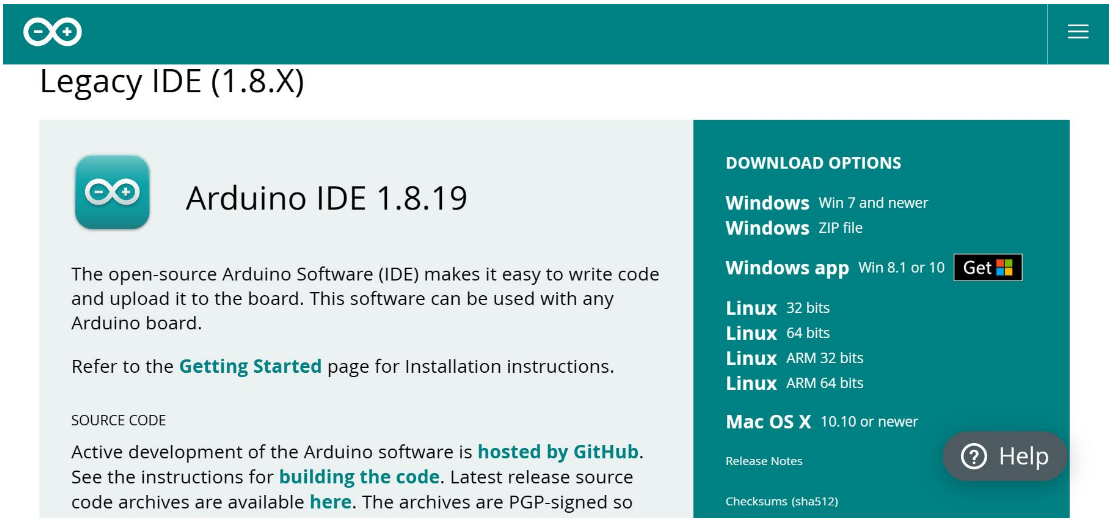
Project files can be obtained from https://github.com/Onyeka-Aribeana/Project-Fingerprinted. The files in this repository include:

- The frontend of the project (in a folder called fingerprinted-frontend)
- The backend of the project (in a folder called fingerprinted-backend)
- The Arduino code (in a folder called arduino_code)
- The database file name, fingerprinted_empty, with the extension .sql

INSTALLATION GUIDE

To get the software up and running the required applications would need to be downloaded (based on your operating system) first.

STEP 1: Download and Install Arduino IDE (Legacy edition) from Software | Arduino. This is used to make required edits to the Arduino code and upload these changes to the ESP8266 module. Note: The legacy edition is used due to USB bus compatibility issues with the newer IDE software. Figure 1. below shows the Arduino IDE version with download options for multiple operating systems.



Legacy IDE (1.8.X)

Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the **Getting Started** page for Installation instructions.

SOURCE CODE

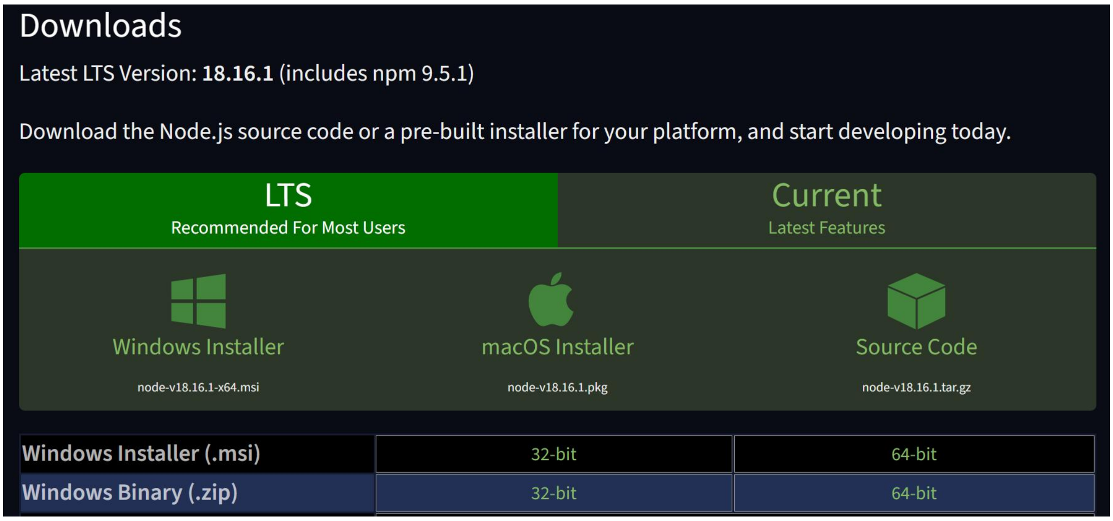
Active development of the Arduino software is **hosted by GitHub**. See the instructions for **building the code**. Latest release source code archives are available **here**. The archives are PGP-signed so

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file
Windows app Win 8.1 or 10 Get
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits
Mac OS X 10.10 or newer
Release Notes
Checksums (sha512)

Figure 1: Arduino Legacy IDE on https://www.arduino.cc/en/software

STEP 2: Download and install Node.js LTS (Long Time Support) for package installation. This would be used to install the package files and dependencies for the front end of the project. Figure 2 below depicts the various versions of Node.js for various operating systems.



Downloads

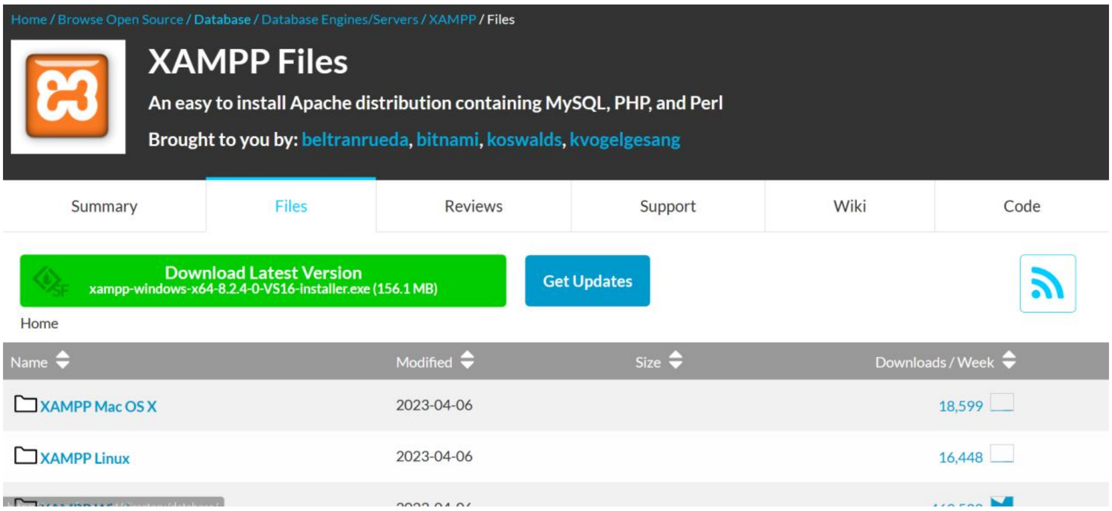
Latest LTS Version: **18.16.1** (includes npm 9.5.1)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit

Figure 2: Node.js package installer on https://nodejs.org/en/download

STEP 3: Download and install XAMPP from Apache. This is used to serve the database conversing with the back-end locally. Figure 3 below shows the versions of xampp for the various operating systems



Home / Browse Open Source / Database / Database Engines/Servers / XAMPP / Files

XAMPP Files

An easy to install Apache distribution containing MySQL, PHP, and Perl

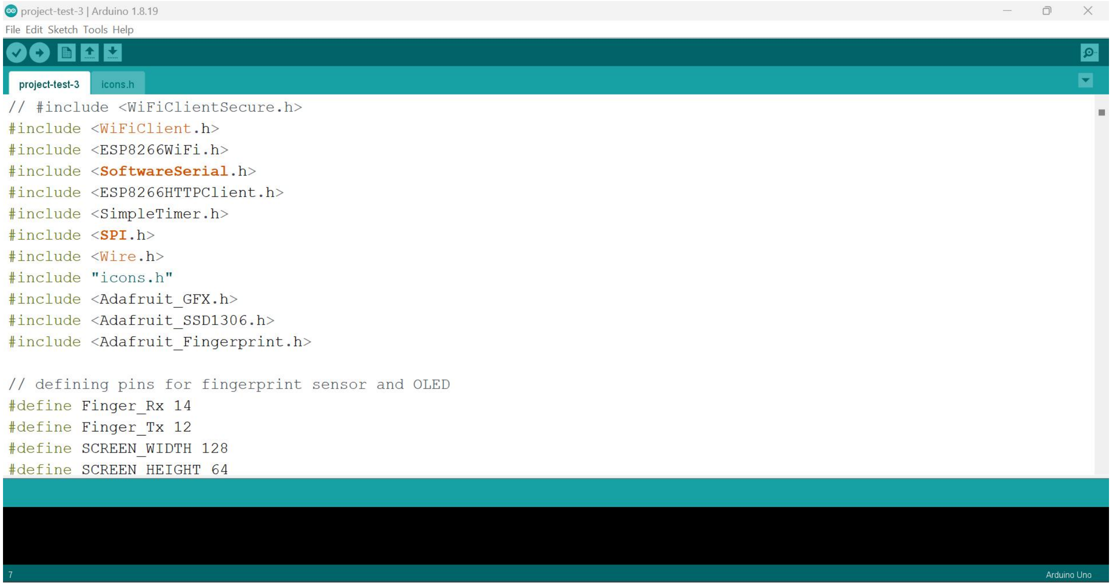
Brought to you by: beltranrueda, bitnami, koswalds, kvogelgesang

Name	Modified	Size	Downloads / Week
XAMPP Mac OS X	2023-04-06		18,599
XAMPP Linux	2023-04-06		16,448

Figure 3: XAMPP on https://sourceforge.net/projects/xampp/files/

SETTING UP THE ARDUINO FILE AND INSTALLING THE REQUIRED LIBRARIES

STEP 1: Open the Arduino IDE application. Go to the toolbar and click on File >> Open…>> {go to the location of arduino_code on your system and open the file 'project-test-3'. The following window will be seen.



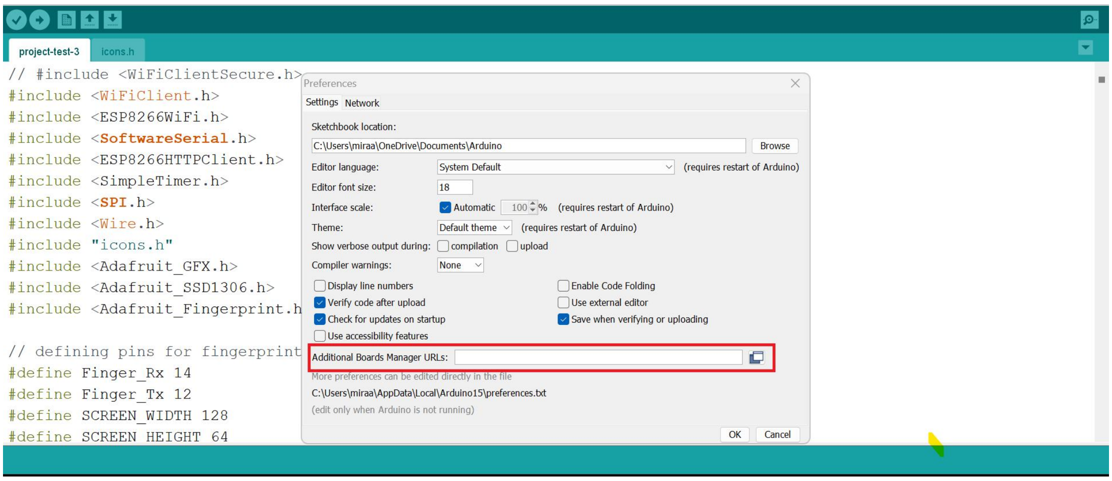
```
// #include <WiFiClientSecure.h>
#include <WiFiClient.h>
#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>
#include <ESP8266HTTPClient.h>
#include <SimpleTimer.h>
#include <SPI.h>
#include <Wire.h>
#include "icons.h"
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_Fingerprint.h>

// defining pins for fingerprint sensor and OLED
#define Finger_Rx 14
#define Finger_Tx 12
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
```

Figure 4: Project code opened on the Arduino integrated development environment.

STEP 2: Go to File >> Preferences or just click Ctrl + comma. You would see the following window. At the Additional Board Managers URL field, which is the field highlighted in red, include the following links separated by a comma.

- For ESP8266 and newer NodeMCU boards: http://arduino.esp8266.com/stable/package_esp8266com_index.json

- For ESP32: https://dl.espressif.com/dl/package_esp32_index.json



```
// #include <WiFiClientSecure.h>
#include <WiFiClient.h>
#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>
#include <ESP8266HTTPClient.h>
#include <SimpleTimer.h>
#include <SPI.h>
#include <Wire.h>
#include "icons.h"
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_Fingerprint.h

// defining pins for fingerprint
#define Finger_Rx 14
#define Finger_Tx 12
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
```

Sketchbook location: C:\Users\miraa\OneDrive\Documents\Arduino

Editor language: System Default (requires restart of Arduino)

Editor font size: 18

Interface scale: ☑ Automatic 100 % (requires restart of Arduino)

Theme: Default theme (requires restart of Arduino)

Show verbose output during: ☐ compilation ☐ upload

Compiler warnings: None

☐ Display line numbers ☐ Enable Code Folding

☑ Verify code after upload ☐ Use external editor

☑ Check for updates on startup ☑ Save when verifying or uploading

☐ Use accessibility features

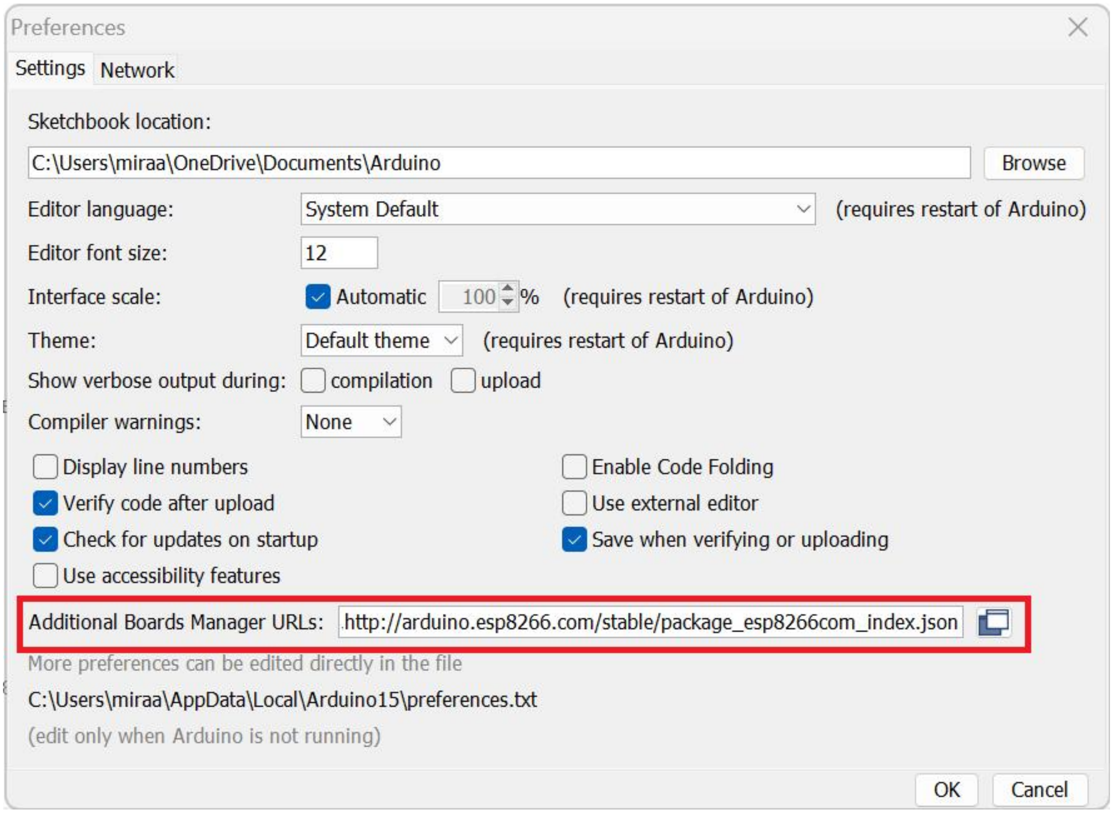
Additional Boards Manager URLs:

More preferences can be edited directly in the file

C:\Users\miraa\AppData\Local\Arduino15\preferences.txt

(edit only when Arduino is not running)

Figure 5: Preferences tab on Arduino IDE



Preferences

Settings Network

Sketchbook location: C:\Users\miraa\OneDrive\Documents\Arduino [Browse]

Editor language: System Default (requires restart of Arduino)

Editor font size: 12

Interface scale: ☑ Automatic 100 % (requires restart of Arduino)

Theme: Default theme (requires restart of Arduino)

Show verbose output during: □ compilation □ upload

Compiler warnings: None

□ Display line numbers
☑ Verify code after upload
☑ Check for updates on startup
□ Use accessibility features

□ Enable Code Folding
□ Use external editor
☑ Save when verifying or uploading

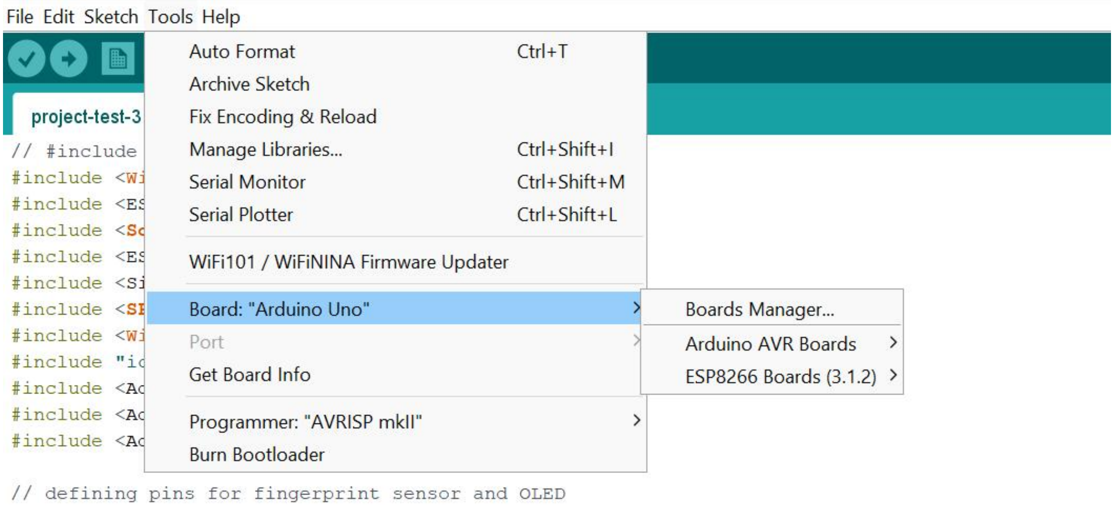
Additional Boards Manager URLs: http://arduino.esp8266.com/stable/package_esp8266com_index.json

More preferences can be edited directly in the file
C:\Users\miraa\AppData\Local\Arduino15\preferences.txt
(edit only when Arduino is not running)

[OK] [Cancel]

Figure 6: Preferences tab with the links for the additional boards

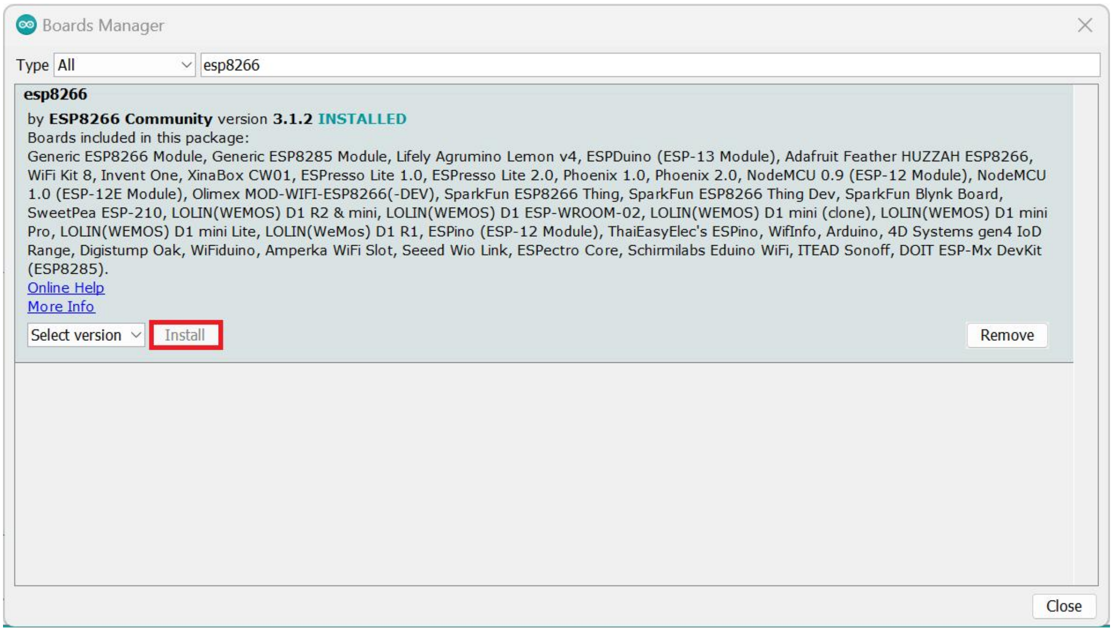
STEP 3: On the toolbar, Go to Tools >> Board: Arduino Uno >> Boards Manager.



```
// #include
#include <Wi
#include <ES
#include <So
#include <ES
#include <Si
#include <SP
#include <Wi
#include "id
#include <Ad
#include <Ad
#include <Ad

// defining pins for fingerprint sensor and OLED
```

Figure 7: Toolbar navigation to Boards Manager

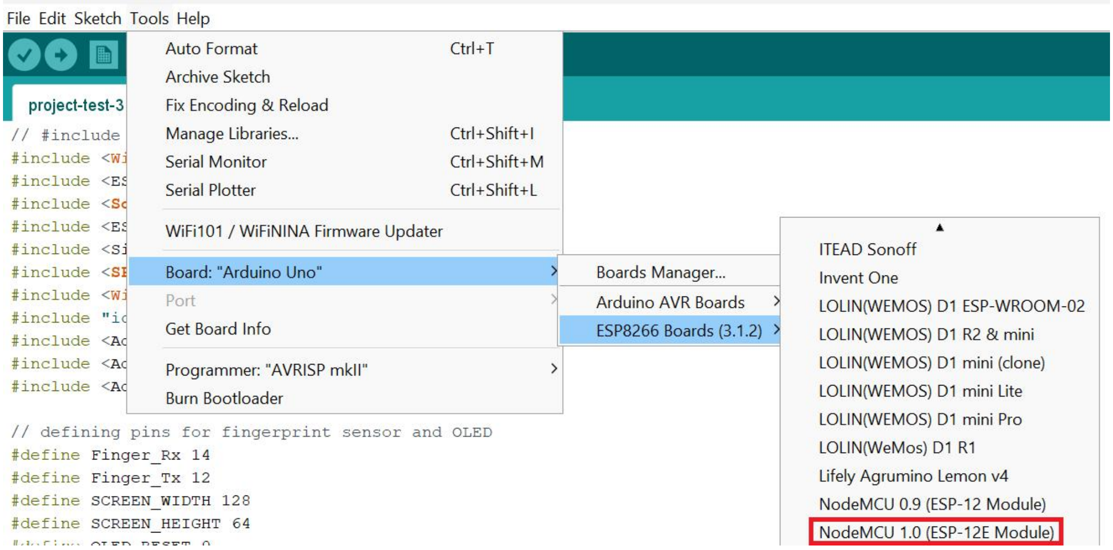
Using the search bar, search for the esp8266 board library and install the board.



esp8266
by **ESP8266 Community** version **3.1.2** **INSTALLED**
Boards included in this package:
Generic ESP8266 Module, Generic ESP8285 Module, Lifely Agrumino Lemon v4, ESPDuino (ESP-13 Module), Adafruit Feather HUZZAH ESP8266, WiFi Kit 8, Invent One, XinaBox CW01, ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Olimex MOD-WIFI-ESP8266(-DEV), SparkFun ESP8266 Thing, SparkFun ESP8266 Thing Dev, SparkFun Blynk Board, SweetPea ESP-210, LOLIN(WEMOS) D1 R2 & mini, LOLIN(WEMOS) D1 ESP-WROOM-02, LOLIN(WEMOS) D1 mini (clone), LOLIN(WEMOS) D1 mini Pro, LOLIN(WEMOS) D1 mini Lite, LOLIN(WeMos) D1 R1, ESPino (ESP-12 Module), ThaiEasyElec's ESPino, WifInfo, Arduino, 4D Systems gen4 IoD Range, Digistump Oak, WiFiduino, Amperka WiFi Slot, Seeed Wio Link, ESPectro Core, Schirmilabs Eduino WiFi, ITEAD Sonoff, DOIT ESP-Mx DevKit (ESP8285).
Online Help
More Info

Figure 8: ESP8266 board installed on boards manager.

Go to Tools >> Board: Arduino Uno >> ESP8266 Board (3.1.2) >> NodeMCU 1.0 (ESP-12E Module)

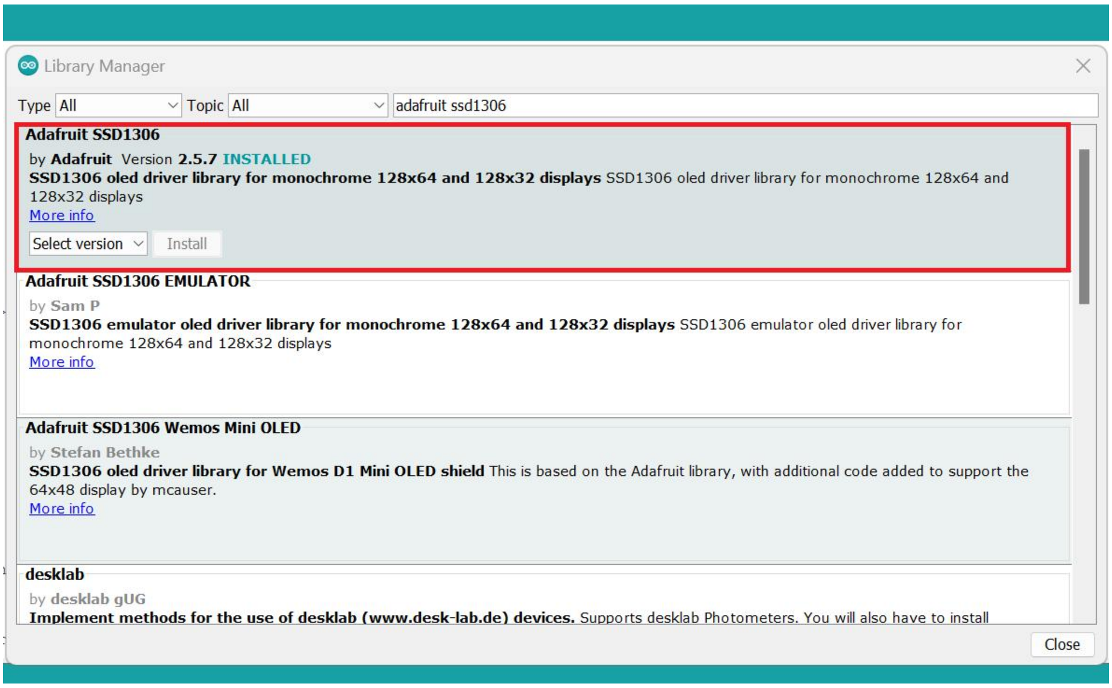

File Edit Sketch Tools Help
project-test-3
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "Arduino Uno"
Port
Get Board Info
Programmer: "AVRISP mkII"
Burn Bootloader
Boards Manager...
Arduino AVR Boards
ESP8266 Boards (3.1.2)
ITEAD Sonoff
Invent One
LOLIN(WEMOS) D1 ESP-WROOM-02
LOLIN(WEMOS) D1 R2 & mini
LOLIN(WEMOS) D1 mini (clone)
LOLIN(WEMOS) D1 mini Lite
LOLIN(WEMOS) D1 mini Pro
LOLIN(WeMos) D1 R1
Lifely Agrumino Lemon v4
NodeMCU 0.9 (ESP-12 Module)
NodeMCU 1.0 (ESP-12E Module)

```
// #include
#include <Wi
#include <ES
#include <Sc
#include <ES
#include <Si
#include <SI
#include <Wi
#include "ic
#include <Ac
#include <Ac
#include <Ac

// defining pins for fingerprint sensor and OLED
#define Finger_Rx 14
#define Finger_Tx 12
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
```

Figure 9: Selecting the ESP8266 board on Arduino IDE

STEP 4: Go to Tools >> Manage Libraries or Ctrl + Shirt + I. A tab called Manage Libraries will be opened. The following libraries would need to be downloaded for the ESP8266 module to be able to manipulate the other components:

- Adafruit SSD1306



Type All Topic All adafruit ssd1306

Adafruit SSD1306
by **Adafruit** Version **2.5.7** **INSTALLED**
SSD1306 oled driver library for monochrome 128x64 and 128x32 displays SSD1306 oled driver library for monochrome 128x64 and 128x32 displays
More info

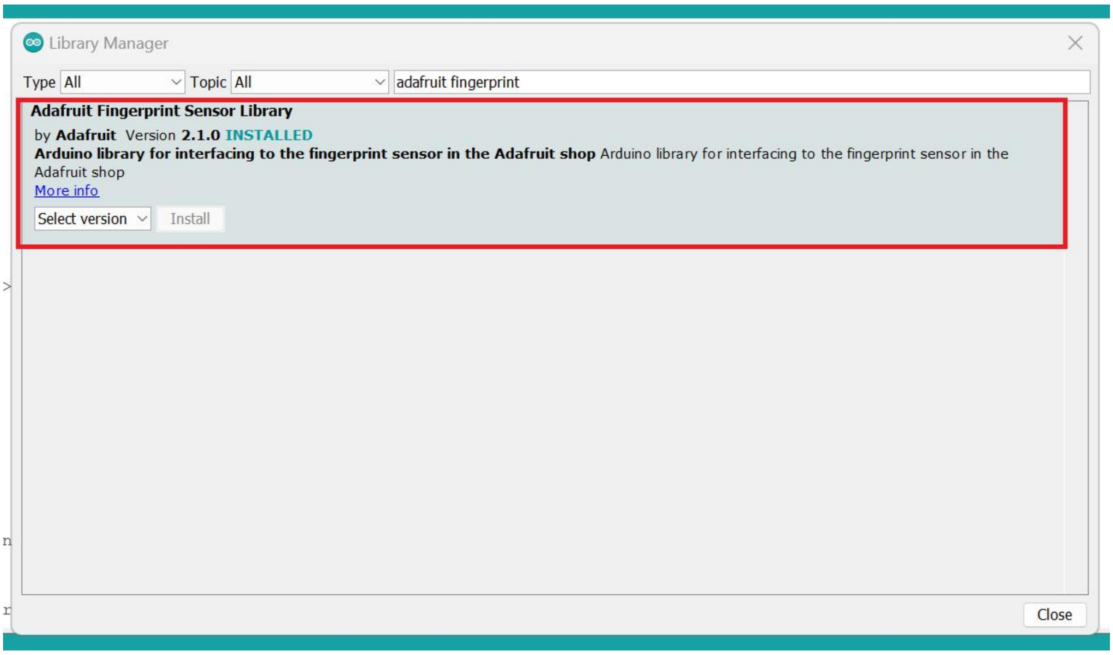
Adafruit SSD1306 EMULATOR
by **Sam P**
SSD1306 emulator oled driver library for monochrome 128x64 and 128x32 displays SSD1306 emulator oled driver library for monochrome 128x64 and 128x32 displays
More info

Adafruit SSD1306 Wemos Mini OLED
by **Stefan Bethke**
SSD1306 oled driver library for Wemos D1 Mini OLED shield This is based on the Adafruit library, with additional code added to support the 64x48 display by mcauser.
More info

desklab
by **desklab gUG**
Implement methods for the use of desklab (www.desk-lab.de) devices. Supports desklab Photometers. You will also have to install

Figure 10: Adafruit SSD1306 on Library Manager

- Adafruit Fingerprint Sensor Library

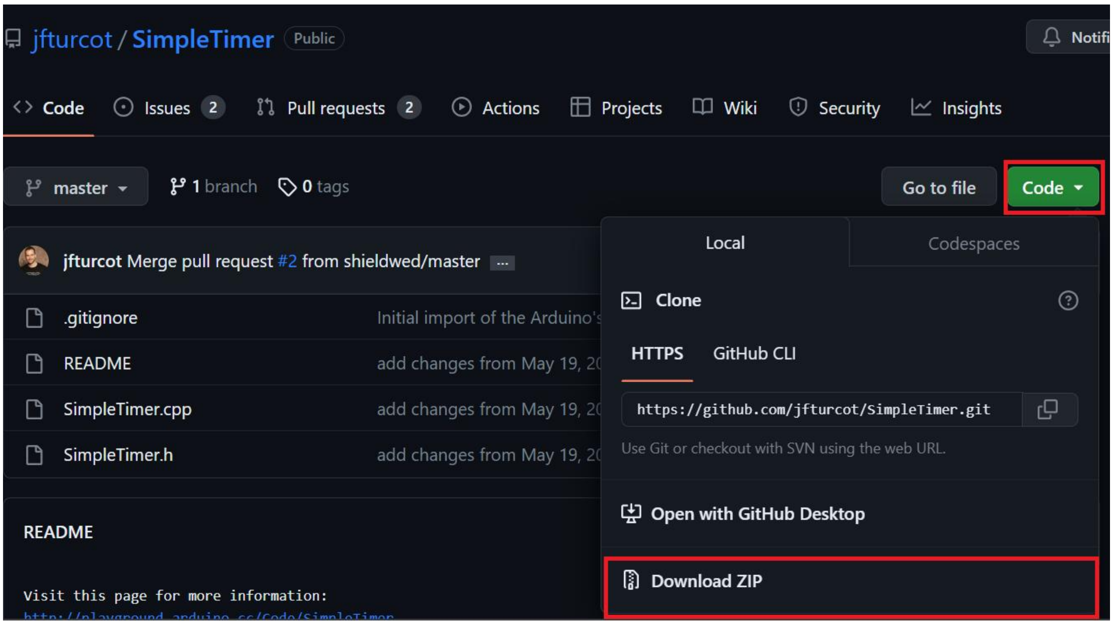


Type All Topic All adafruit fingerprint

Adafruit Fingerprint Sensor Library
by **Adafruit** Version **2.1.0** **INSTALLED**
Arduino library for interfacing to the fingerprint sensor in the Adafruit shop Arduino library for interfacing to the fingerprint sensor in the Adafruit shop
More info

Figure 11: Adafruit fingerprint sensor library on Library Manager

STEP 5: The last library, Simple Timer by jfurcot would need to be installed manually. Go to https://github.com/jfurcot/SimpleTimer on your browser. Select Code >> Download ZIP



jfturcot Merge pull request #2 from shieldwed/master ...

.gitignore	Initial import of the Arduino's
README	add changes from May 19, 20
SimpleTimer.cpp	add changes from May 19, 20
SimpleTimer.h	add changes from May 19, 20

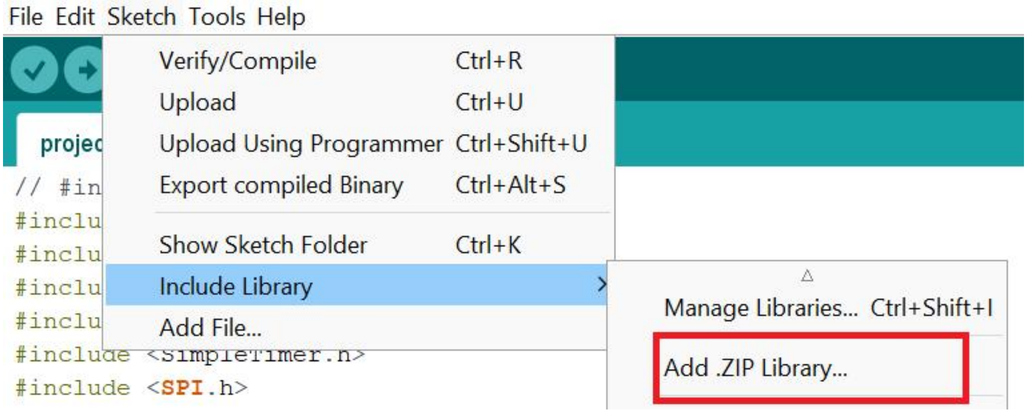
https://github.com/jfturcot/SimpleTimer.git

Use Git or checkout with SVN using the web URL.

Visit this page for more information:

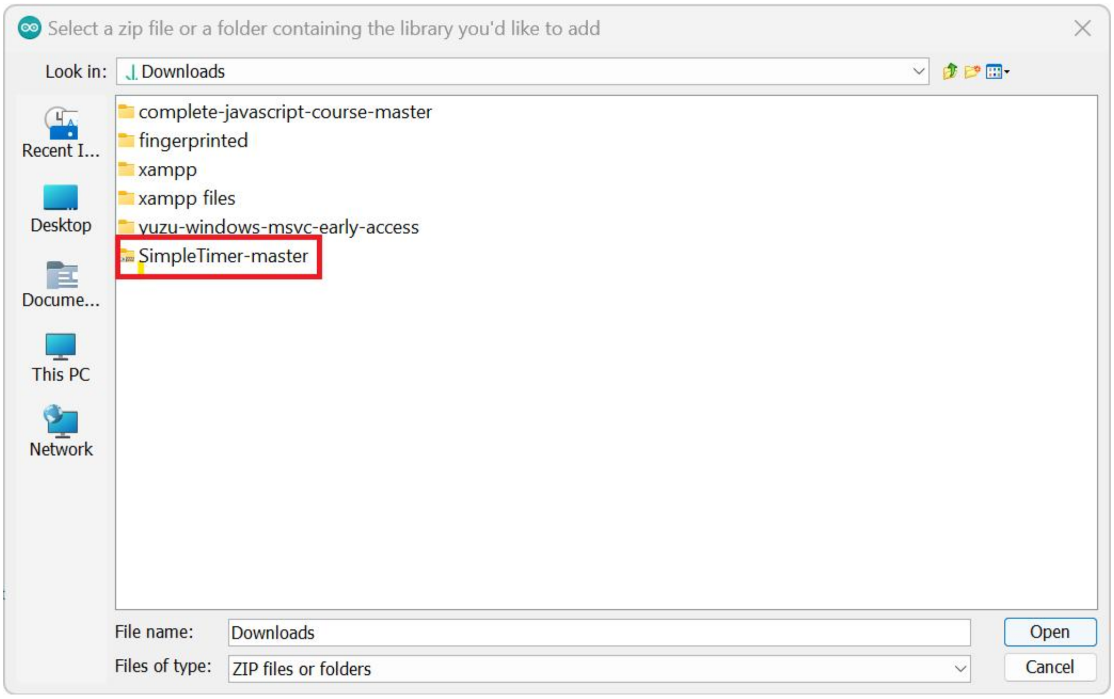
Figure 12: Downloading the zip file of the simple timer library on GitHub

On Arduino IDE, Go to Sketch >> Include Library >> Add .ZIP library. Locate the SimpleTimer-master file in your downloads and select it


File Edit Sketch Tools Help
projec
Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library
Add File...
Manage Libraries... Ctrl+Shift+I
Add .ZIP Library...

```
// #in
#inclu
#inclu
#inclu
#inclu
#include <SimpleTimer.h>
#include <SPI.h>
```

Figure 13: Manual addition of an Arduino library



Select a zip file or a folder containing the library you'd like to add

Look in: Downloads

- complete-javascript-course-master
- fingerprinted
- xampp
- xampp files
- yuzu-windows-msvc-early-access
- SimpleTimer-master

File name: Downloads

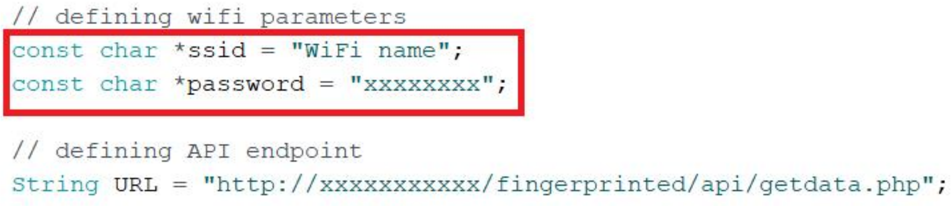
Files of type: ZIP files or folders

Figure 14: Window for selecting the .ZIP file

CHANGING THE NETWORK PARAMETERS

In the code, a few variables would need to be changed to correspond with your WiFi and network settings.

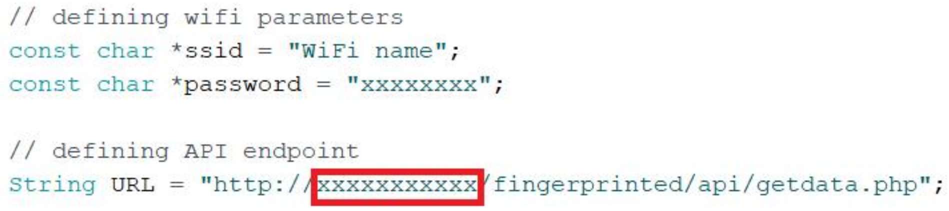
STEP 1: The variable ssid and password in the code would need to be changed to your WiFi name and password respectively.



```
// defining wifi parameters
const char *ssid = "WiFi name";
const char *password = "xxxxxxxx";

// defining API endpoint
String URL = "http://xxxxxxxxxxx/fingerprinted/api/getdata.php";
```

Figure 15: WiFi ssid and password highlighted in red

STEP 2: The IPv4 address of the WiFi network would replace the unknown in the API endpoint URL highlighted in red in the figure below. You can get your IPv4 address by checking the network settings for the connected WiFi network settings.



```
// defining wifi parameters
const char *ssid = "WiFi name";
const char *password = "xxxxxxxx";

// defining API endpoint
String URL = "http://xxxxxxxxxxx/fingerprinted/api/getdata.php";
```

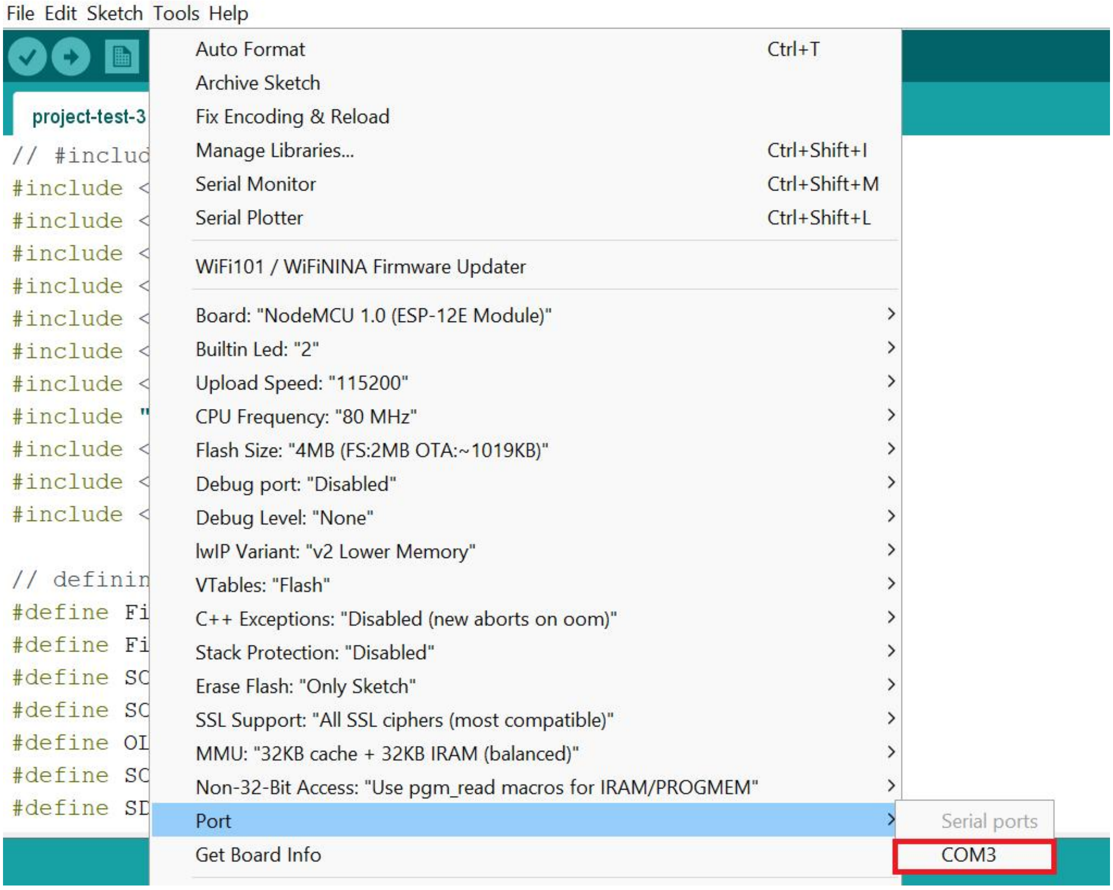
Figure 16: API endpoint URL IP address highlighted

Note: The WiFi used in the code has to be the same one used on the computer serving the database locally.

UPLOADING CODE TO THE ESP8266 MODULE

STEP 1: Connect the ESP8266 to the computer using the USB port.

STEP 2: Go to Tools >> Port and select the USB port connected to the ESP8266.



File Edit Sketch Tools Help
project-test-3
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Port
Get Board Info
Serial ports
COM3

Board: "NodeMCU 1.0 (ESP-12E Module)"
Builtin Led: "2"
Upload Speed: "115200"
CPU Frequency: "80 MHz"
Flash Size: "4MB (FS:2MB OTA:~1019KB)"
Debug port: "Disabled"
Debug Level: "None"
lwIP Variant: "v2 Lower Memory"
VTables: "Flash"
C++ Exceptions: "Disabled (new aborts on oom)"
Stack Protection: "Disabled"
Erase Flash: "Only Sketch"
SSL Support: "All SSL ciphers (most compatible)"
MMU: "32KB cache + 32KB IRAM (balanced)"
Non-32-Bit Access: "Use pgm_read macros for IRAM/PROGMEM"

```
// #incluc
#include <
#include <
#include <
#include <
#include <
#include <
#include <
#include "
#include <
#include <
#include <

// definin
#define Fi
#define Fi
#define SC
#define SC
#define OI
#define SC
#define SI
```

Figure 17: Selecting the USB port for upload

STEP 3: Upload the code to the ESP8266 using the upload button.

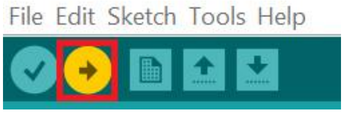
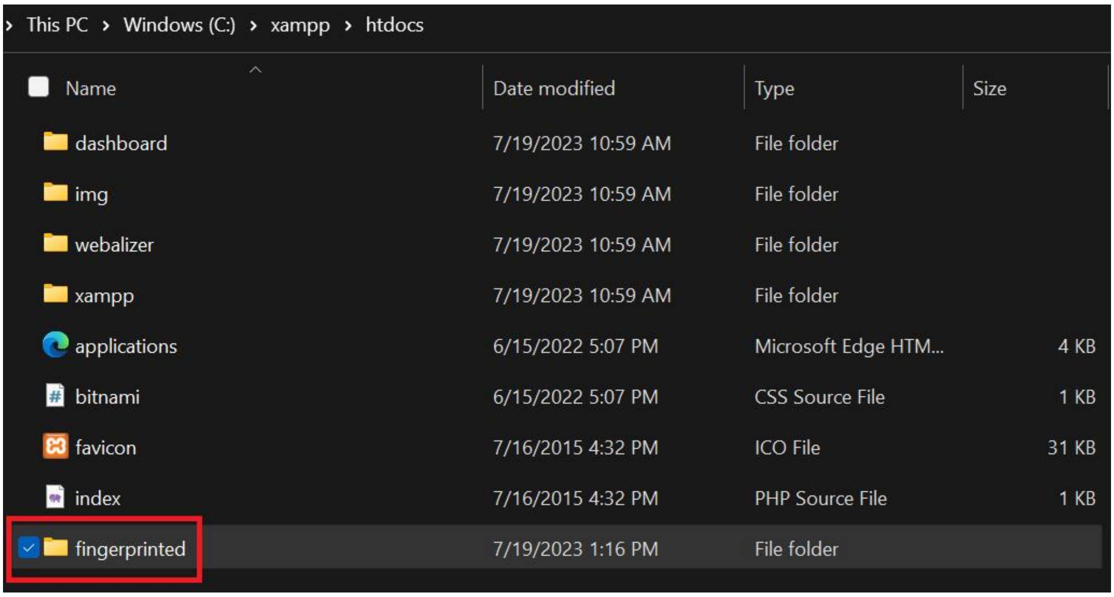


Figure 18: Upload button highlighted in red

SETTING UP THE BACKEND OF THE WEBSITE

To set up the back-end of the website, we need to do a few things with the database hosting platform, XAMPP.

STEP 1: Locate the xampp folder on your computer. On installation, by default, it is found in the root folder of your computer which is C:\. If changed to a different location on installation, then use the new location. Go to the htdocs folder and copy and paste the back-end folder (fingerprinted-backend) in the htdocs folder.



› This PC › Windows (C:) › xampp › htdocs

Name	Date modified	Type	Size
dashboard	7/19/2023 10:59 AM	File folder	
img	7/19/2023 10:59 AM	File folder	
webalizer	7/19/2023 10:59 AM	File folder	
xampp	7/19/2023 10:59 AM	File folder	
applications	6/15/2022 5:07 PM	Microsoft Edge HTM...	4 KB
bitnami	6/15/2022 5:07 PM	CSS Source File	1 KB
favicon	7/16/2015 4:32 PM	ICO File	31 KB
index	7/16/2015 4:32 PM	PHP Source File	1 KB
☑ fingerprinted	7/19/2023 1:16 PM	File folder	

Figure 19: fingerprinted back-end folder in htdocs of xampp.

On my computer, the back-end folder is named fingerprinted. The htdocs folder is where the programs for the web pages would be stored.

STEP 2: Run the application XAMPP as an administrator. A window, like the one in Figure 21, will be opened

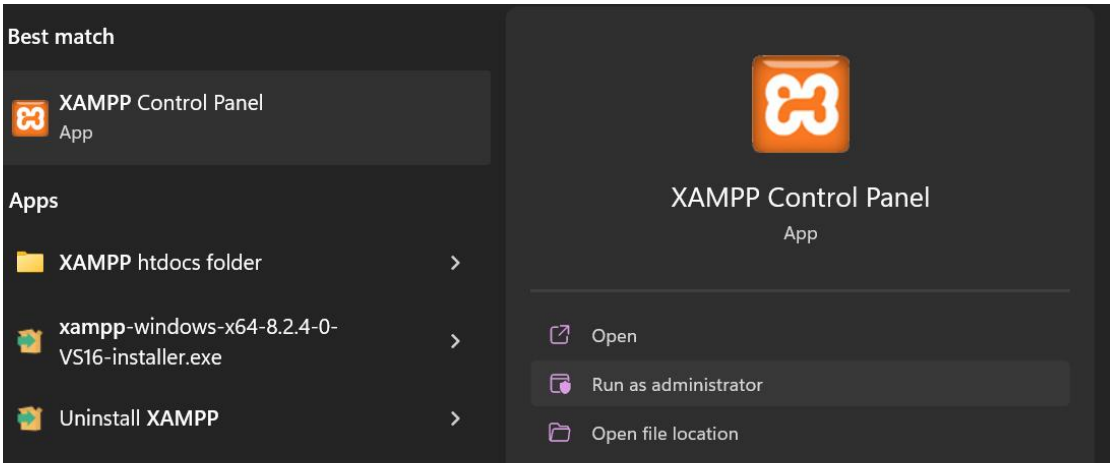
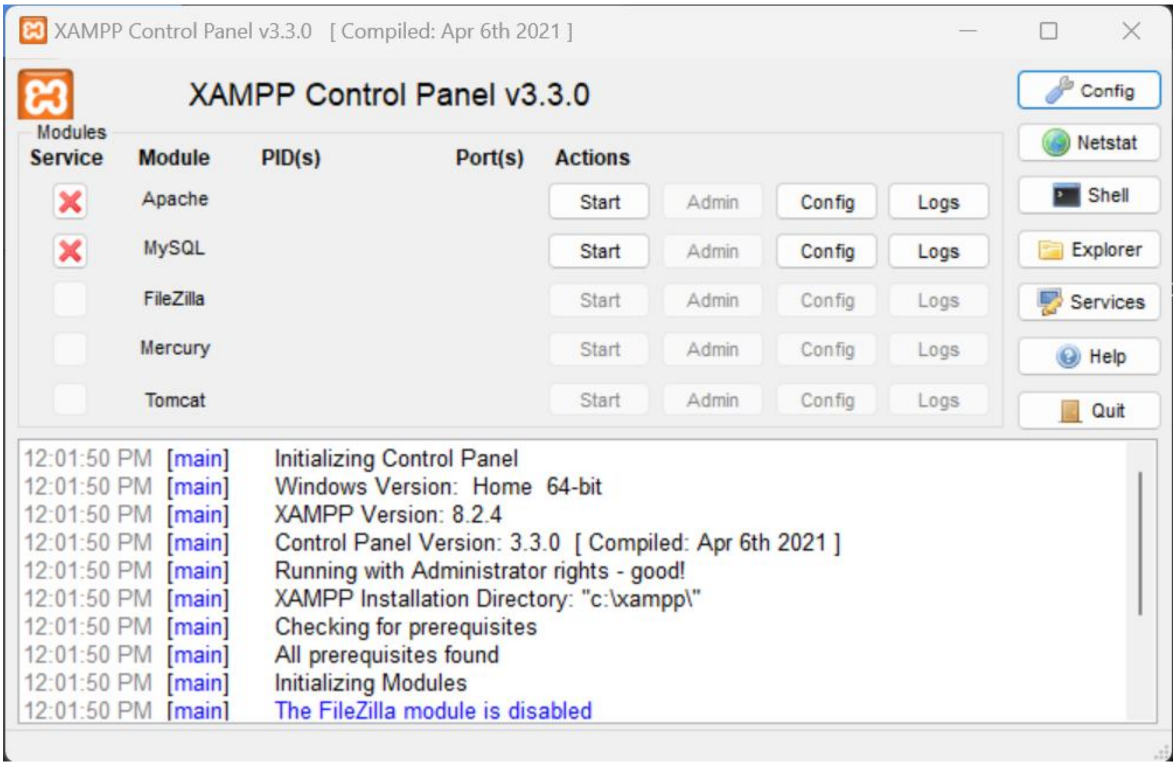


Figure 20: Running XAMPP as administrator

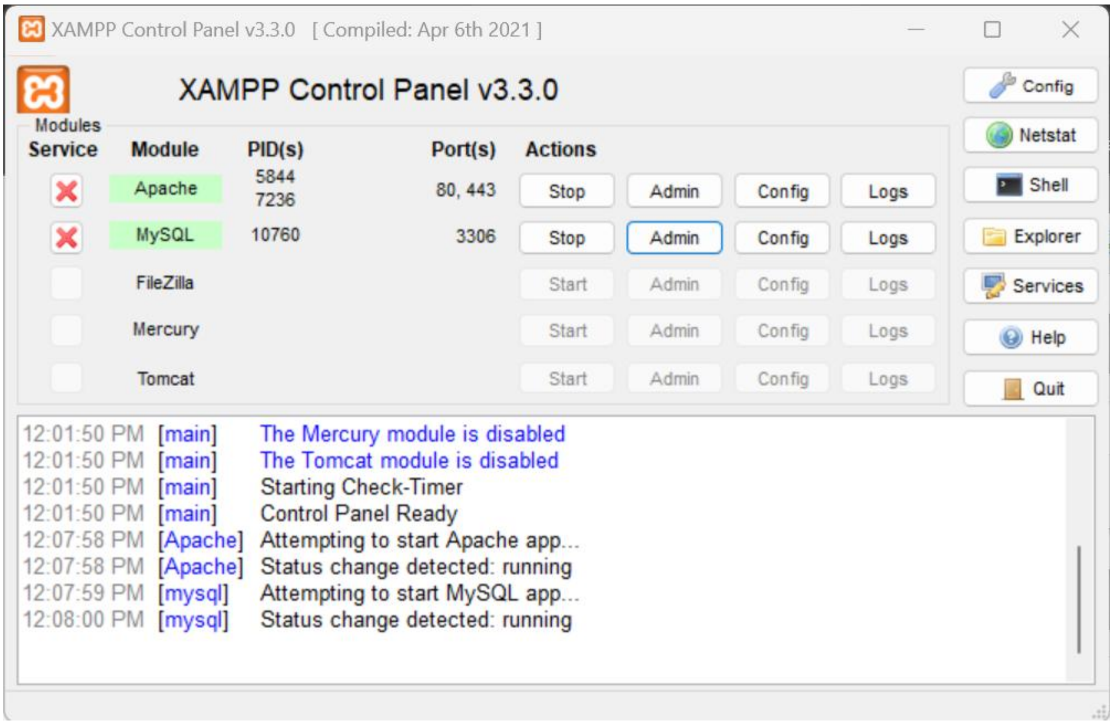

XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]
XAMPP Control Panel v3.3.0
Modules
Config
Netstat
Shell
Explorer
Services
Help
Quit

Service	Module	PID(s)	Port(s)	Actions
☒	Apache			Start Admin Config Logs
☒	MySQL			Start Admin Config Logs
□	FileZilla			Start Admin Config Logs
□	Mercury			Start Admin Config Logs
□	Tomcat			Start Admin Config Logs

```
12:01:50 PM  [main]    Initializing Control Panel
12:01:50 PM  [main]    Windows Version:  Home  64-bit
12:01:50 PM  [main]    XAMPP Version: 8.2.4
12:01:50 PM  [main]    Control Panel Version: 3.3.0  [ Compiled: Apr 6th 2021 ]
12:01:50 PM  [main]    Running with Administrator rights - good!
12:01:50 PM  [main]    XAMPP Installation Directory: "c:\xampp\"
12:01:50 PM  [main]    Checking for prerequisites
12:01:50 PM  [main]    All prerequisites found
12:01:50 PM  [main]    Initializing Modules
12:01:50 PM  [main]    The FileZilla module is disabled
```

Figure 21: XAMPP control panel

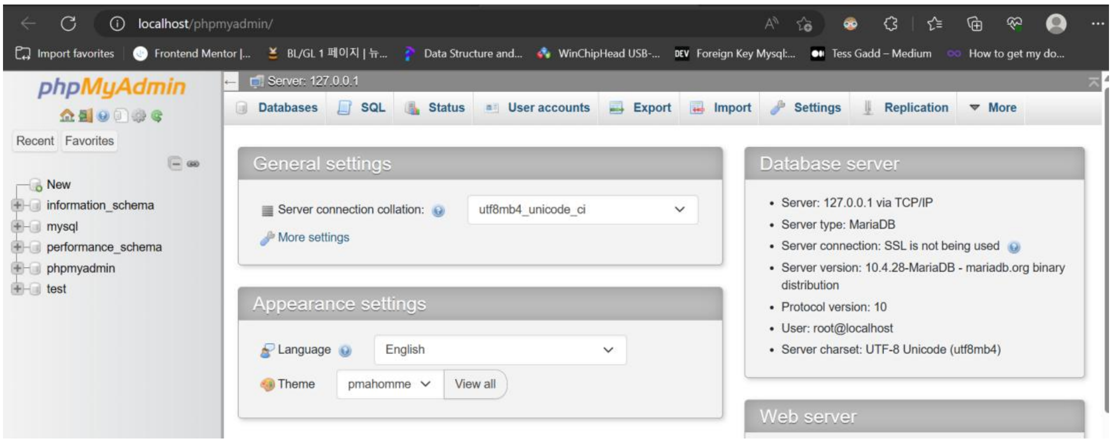
STEP 3: Start the Apache and MySQL servers by clicking the start buttons and open PHPMyAdmin by clicking the Admin button highlighted in blue in the figure below. This action opens PHPMyAdmin on a browser.


XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]
XAMPP Control Panel v3.3.0
Modules
Config
Netstat
Shell
Explorer
Services
Help
Quit

Service	Module	PID(s)	Port(s)	Actions
☒	Apache	5844 7236	80, 443	Stop Admin Config Logs
☒	MySQL	10760	3306	Stop Admin Config Logs
□	FileZilla			Start Admin Config Logs
□	Mercury			Start Admin Config Logs
□	Tomcat			Start Admin Config Logs

```
12:01:50 PM  [main]    The Mercury module is disabled
12:01:50 PM  [main]    The Tomcat module is disabled
12:01:50 PM  [main]    Starting Check-Timer
12:01:50 PM  [main]    Control Panel Ready
12:07:58 PM  [Apache]  Attempting to start Apache app...
12:07:58 PM  [Apache]  Status change detected: running
12:07:59 PM  [mysql]   Attempting to start MySQL app...
12:08:00 PM  [mysql]   Status change detected: running
```

Figure 22: XAMPP control panel with services running



Server connection collation: utf8mb4_unicode_ci

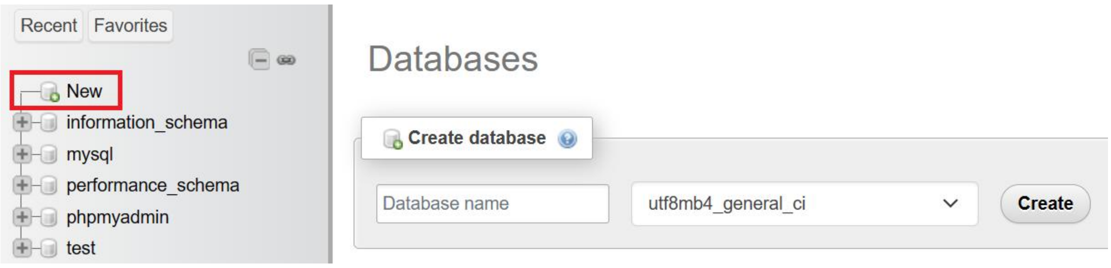
Language: English

Theme: pmahomme

- Server: 127.0.0.1 via TCP/IP
- Server type: MariaDB
- Server connection: SSL is not being used
- Server version: 10.4.28-MariaDB - mariadb.org binary distribution
- Protocol version: 10
- User: root@localhost
- Server charset: UTF-8 Unicode (utf8mb4)

Figure 23: PHPMyAdmin on browser

STEP 4: Click on New highlighted on the left panel. This opens the page for databases, both new and existing.

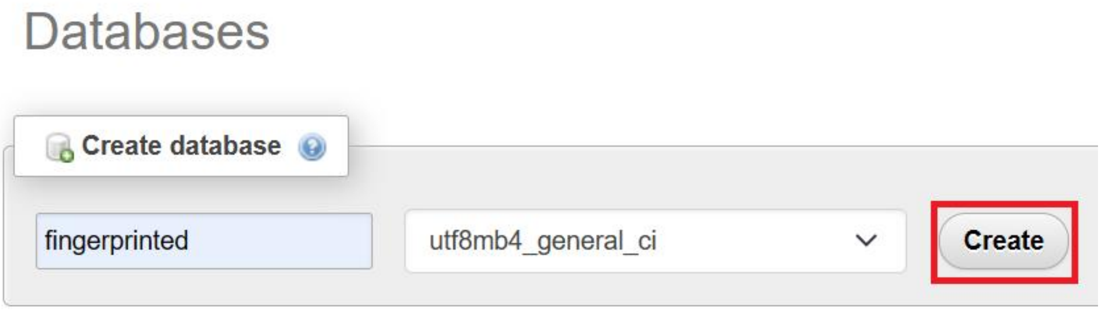


Database name

utf8mb4_general_ci

Figure 24: PHPMyAdmin page to create a new

STEP 5: Create a new database called ‘fingerprinted’ by typing in the field for the database name and clicking the Create button.



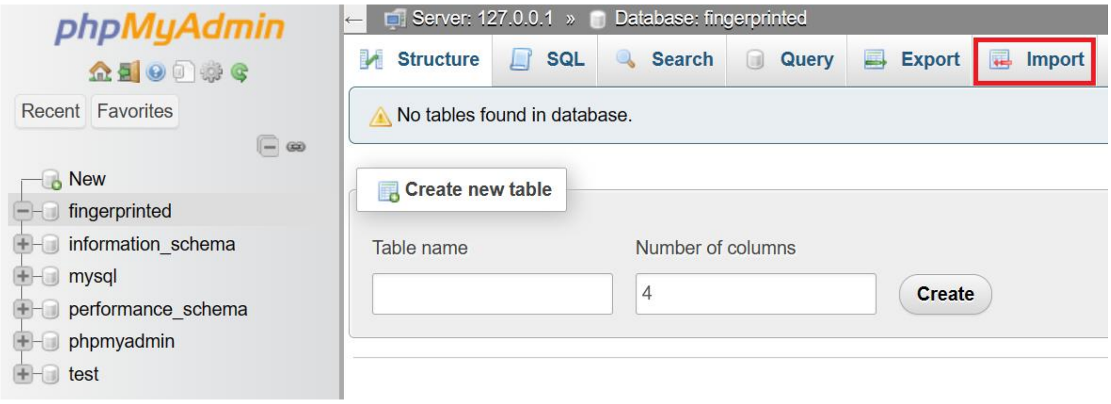
fingerprinted

utf8mb4_general_ci

Figure 25: Creating a new database on PHPMyAdmin

Once created, it appears as a tab on the left side of the panel.

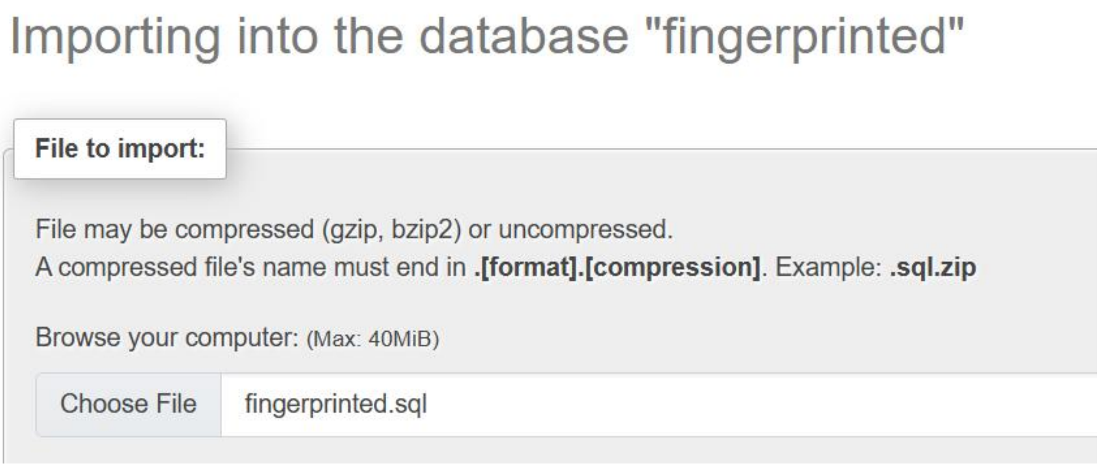
STEP 4: Click the tab ‘fingerprinted’ and click import highlighted on the top bar.



No tables found in database.

Figure 26: fingerprinted database on PHPMyAdmin

STEP 6: Click the field 'Choose File' and choose the database file name fingerprinted_empty.sql on your computer. Once selected, scroll down and click Import to execute.



Importing into the database "fingerprinted"

File to import:

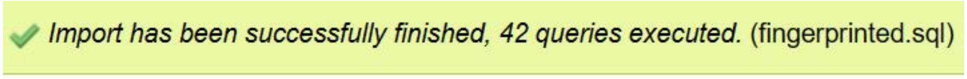
File may be compressed (gzip, bzip2) or uncompressed.
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: (Max: 40MiB)

fingerprinted.sql

Figure 27: Import the database file

You should see a string of success messages like the one in the figure below.



Import has been successfully finished, 42 queries executed. (fingerprinted.sql)

Figure 28: Message on successful importation

STEP 6: Click the tab 'fingerprinted' and click the 'Privileges' tab highlighted on the top bar.

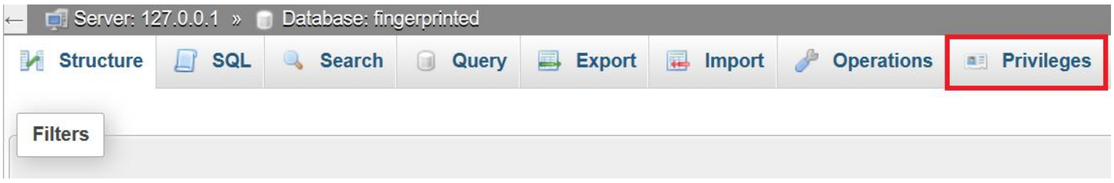
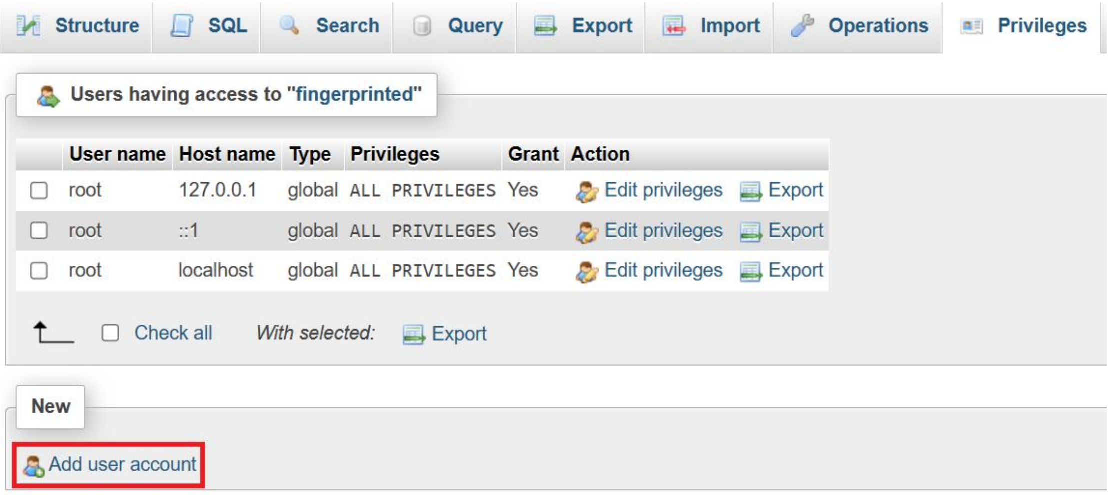


Figure 29: Privileges tab on fingerprinted database

This opens a page for privileges on the database 'fingerprinted'.

STEP 7: Click the ‘Add new account’ highlighted in the figure below



Users having access to "fingerprinted"

	User name	Host name	Type	Privileges	Grant	Action
□	root	127.0.0.1	global	ALL PRIVILEGES	Yes	Edit privileges Export
□	root	::1	global	ALL PRIVILEGES	Yes	Edit privileges Export
□	root	localhost	global	ALL PRIVILEGES	Yes	Edit privileges Export

□ Check all *With selected:* Export

New

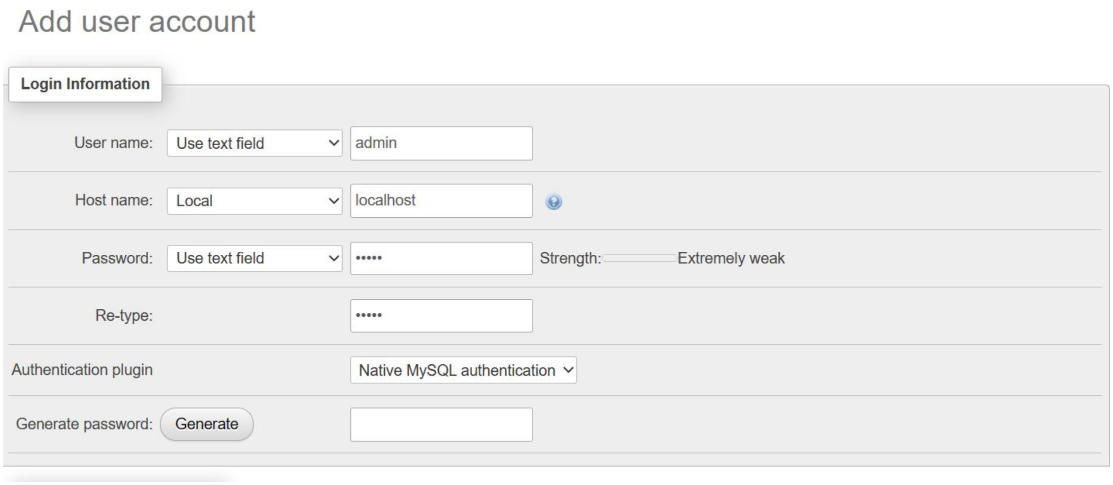
Add user account

Figure 30: Privileges on the fingerprinted database.

This opens a page to add a new account for access to the database. This is required so that the back end would access information from the database.

STEP 8: Fill the following information in the form in the figure below

- User name: admin
- Host name: localhost, you can select the local option in the select input field
- Password: admin



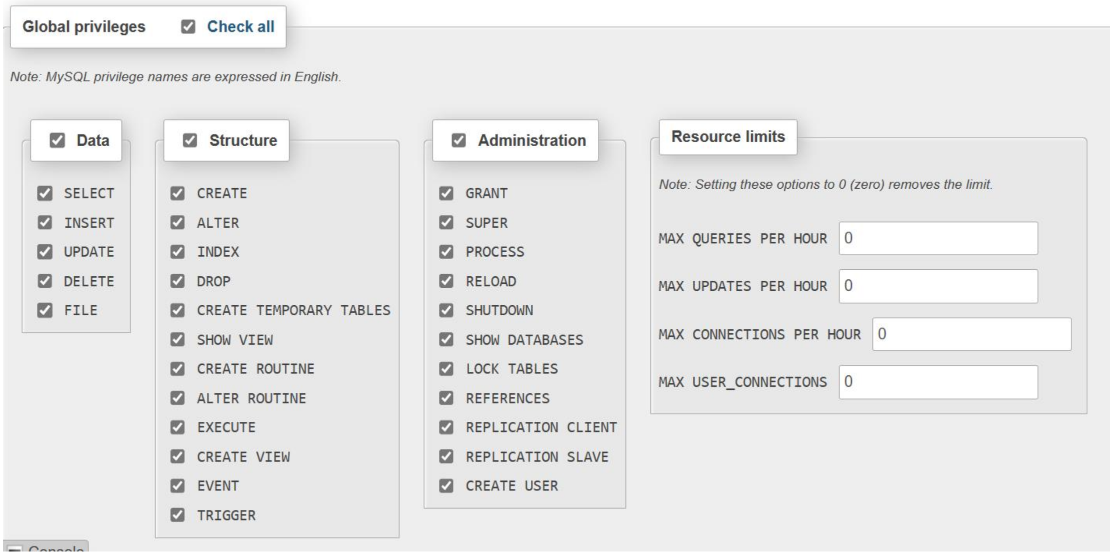
Add user account

Login Information

User name:	Use text field	admin
Host name:	Local	localhost
Password:	Use text field	••••• Strength: Extremely weak
Re-type:		•••••
Authentication plugin		Native MySQL authentication
Generate password:	Generate	

Figure 31: Filled information for a new user account.

STEP 9: Check the ‘Check all’ checkbox for Global privileges.


Global privileges ☑ Check all

Note: MySQL privilege names are expressed in English.

☑ Data
☑ SELECT
☑ INSERT
☑ UPDATE
☑ DELETE
☑ FILE

☑ Structure
☑ CREATE
☑ ALTER
☑ INDEX
☑ DROP
☑ CREATE TEMPORARY TABLES
☑ SHOW VIEW
☑ CREATE ROUTINE
☑ ALTER ROUTINE
☑ EXECUTE
☑ CREATE VIEW
☑ EVENT
☑ TRIGGER

☑ Administration
☑ GRANT
☑ SUPER
☑ PROCESS
☑ RELOAD
☑ SHUTDOWN
☑ SHOW DATABASES
☑ LOCK TABLES
☑ REFERENCES
☑ REPLICATION CLIENT
☑ REPLICATION SLAVE
☑ CREATE USER

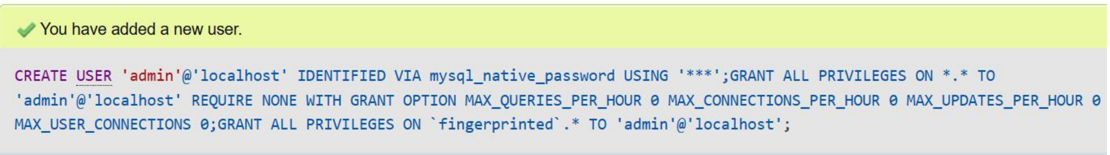
Resource limits

Note: Setting these options to 0 (zero) removes the limit.

MAX QUERIES PER HOUR 0
MAX UPDATES PER HOUR 0
MAX CONNECTIONS PER HOUR 0
MAX USER_CONNECTIONS 0

Figure 32: Global privileges on fingerprinted database

Once concluded, scroll to the end and click the 'Go' button. You should see a success message similar to the one in the figure below.


✓ You have added a new user.

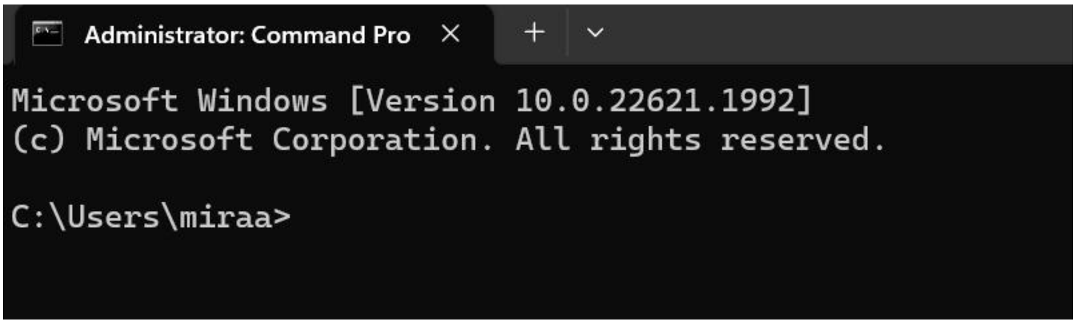
```
CREATE USER 'admin'@'localhost' IDENTIFIED VIA mysql_native_password USING '***';GRANT ALL PRIVILEGES ON *.* TO
'admin'@'localhost' REQUIRE NONE WITH GRANT OPTION MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0;GRANT ALL PRIVILEGES ON `fingerprinted`.* TO 'admin'@'localhost';
```

Figure 33: Message on successful creation of a new user

SETTING UP THE FRONTEND OF THE WEBSITE

To set up the front end of the website, you will need to install a few dependencies on which the website relies to operate properly. A production version of the website would need to be built and locally served for operation on the local database.

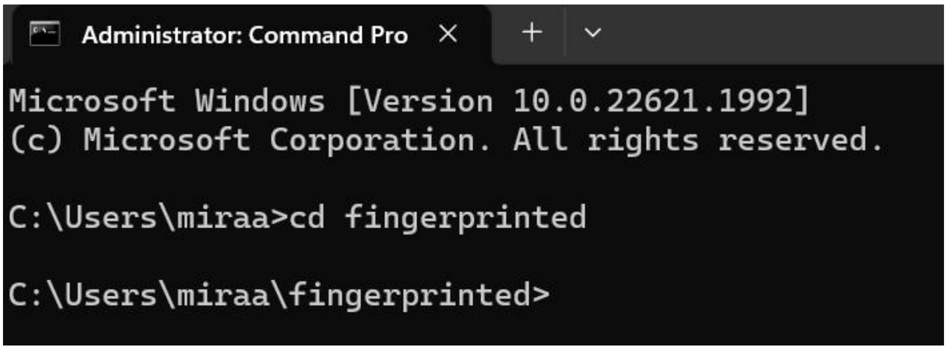
STEP 1: Open the command prompt and navigate to the directory where the front-end folder (fingerprinted-frontend) is located. For example, on my computer, the file is located in my user profile folder which is miraa and its directory is C:\Users\miraa which is the directory opened by default in the command prompt as you can see in the figure below.



```
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\miraa>
```

Figure 34: Command Prompt Window

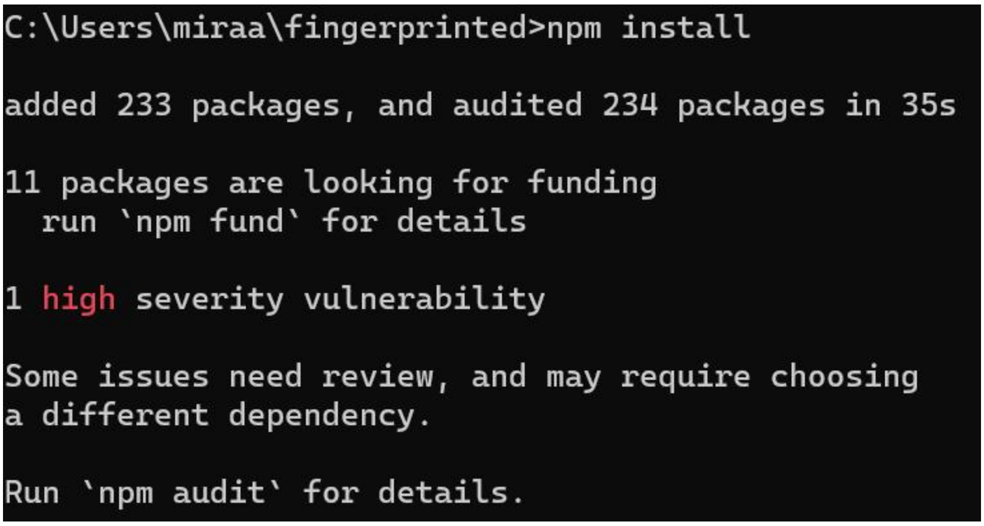
To navigate to the folder, use the command cd (change directory) followed by the folder name. On my computer, the folder is called fingerprinted.



```
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\miraa>cd fingerprinted

C:\Users\miraa\fingerprinted>
```

Figure 35: Navigation to the front-end folder

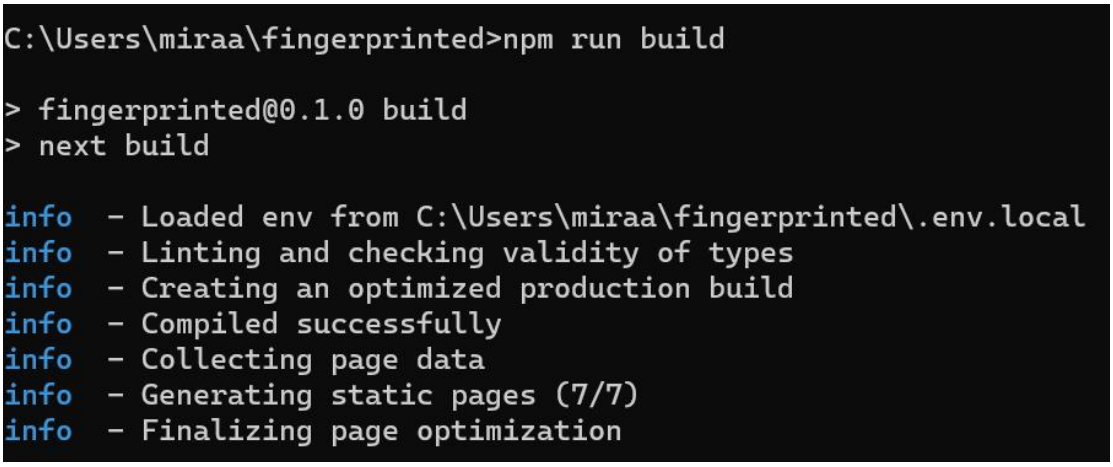
STEP 2: Type in the command ‘npm install’. This command installs all the dependencies required by the front end to work properly.



```
C:\Users\miraa\fingerprinted>npm install

added 233 packages, and audited 234 packages in 35s

11 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

Some issues need review, and may require choosing
a different dependency.

Run `npm audit` for details.
```

Figure 36: Installation of dependency packages on command prompt

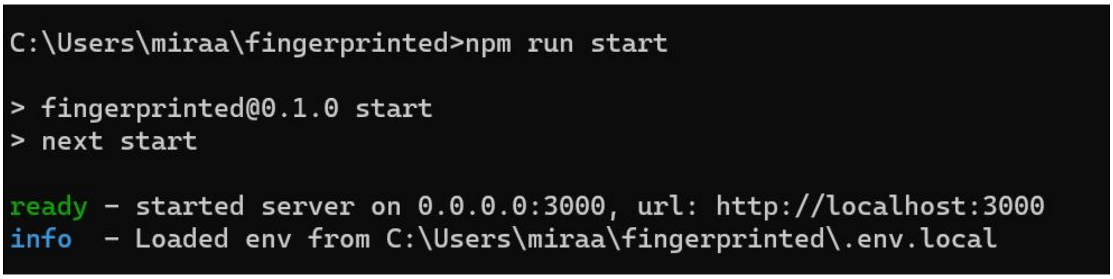
STEP 3: Type in the command ‘npm run build’. This creates a production build of the front end for use.



```
C:\Users\miraa\fingerprinted>npm run build

> fingerprinted@0.1.0 build
> next build

info  - Loaded env from C:\Users\miraa\fingerprinted\.env.local
info  - Linting and checking validity of types
info  - Creating an optimized production build
info  - Compiled successfully
info  - Collecting page data
info  - Generating static pages (7/7)
info  - Finalizing page optimization
```

Figure 37: Creating a production build of the website on command prompt

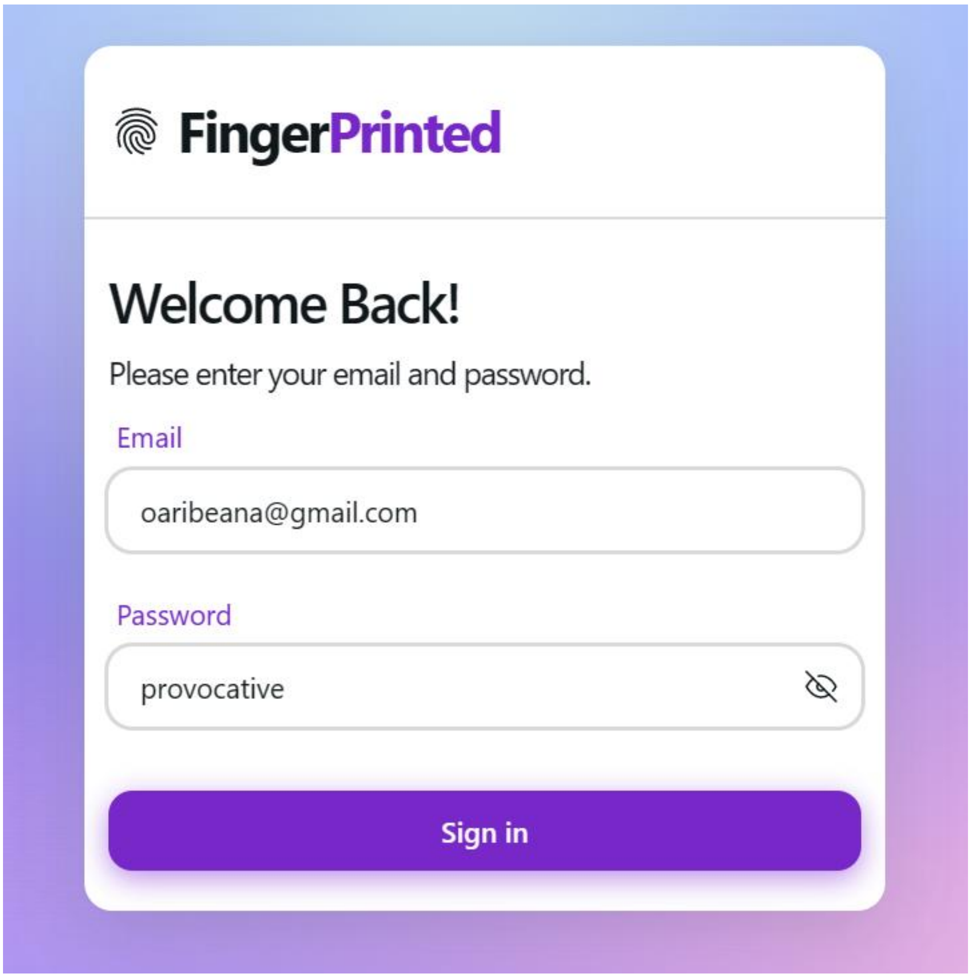
STEP 3: Type in the command 'npm run start'. This initializes the server that renders the production build.



```
C:\Users\miraa\fingerprinted>npm run start

> fingerprinted@0.1.0 start
> next start

ready - started server on 0.0.0.0:3000, url: http://localhost:3000
info  - Loaded env from C:\Users\miraa\fingerprinted\.env.local
```

Figure 38: Starting the server rendering the website on command prompt

STEP 4: Copy the URL: http://localhost:3000 and paste it into your favorite browser, you would see a login screen like the figure shown below. The email and password are provided in the figure below.



Welcome Back!

Please enter your email and password.

Email: oaribeana@gmail.com

Password: provocative

Figure 39: Login page of the website

After filling in login details, you are routed to the index page, http://localhost:3000.