Installation guide:

This installation guide is for the beginner who is completely unfamiliar with using the Arduino IDE.

For users familiar with using the Arduino IDE:
- in 4 the 2 necessary libraries
- in 6 board settings
- in 7 integration in Homey
- Wifi settings are on the first page of the code

Summary :

1- Installing the Arduino IDE
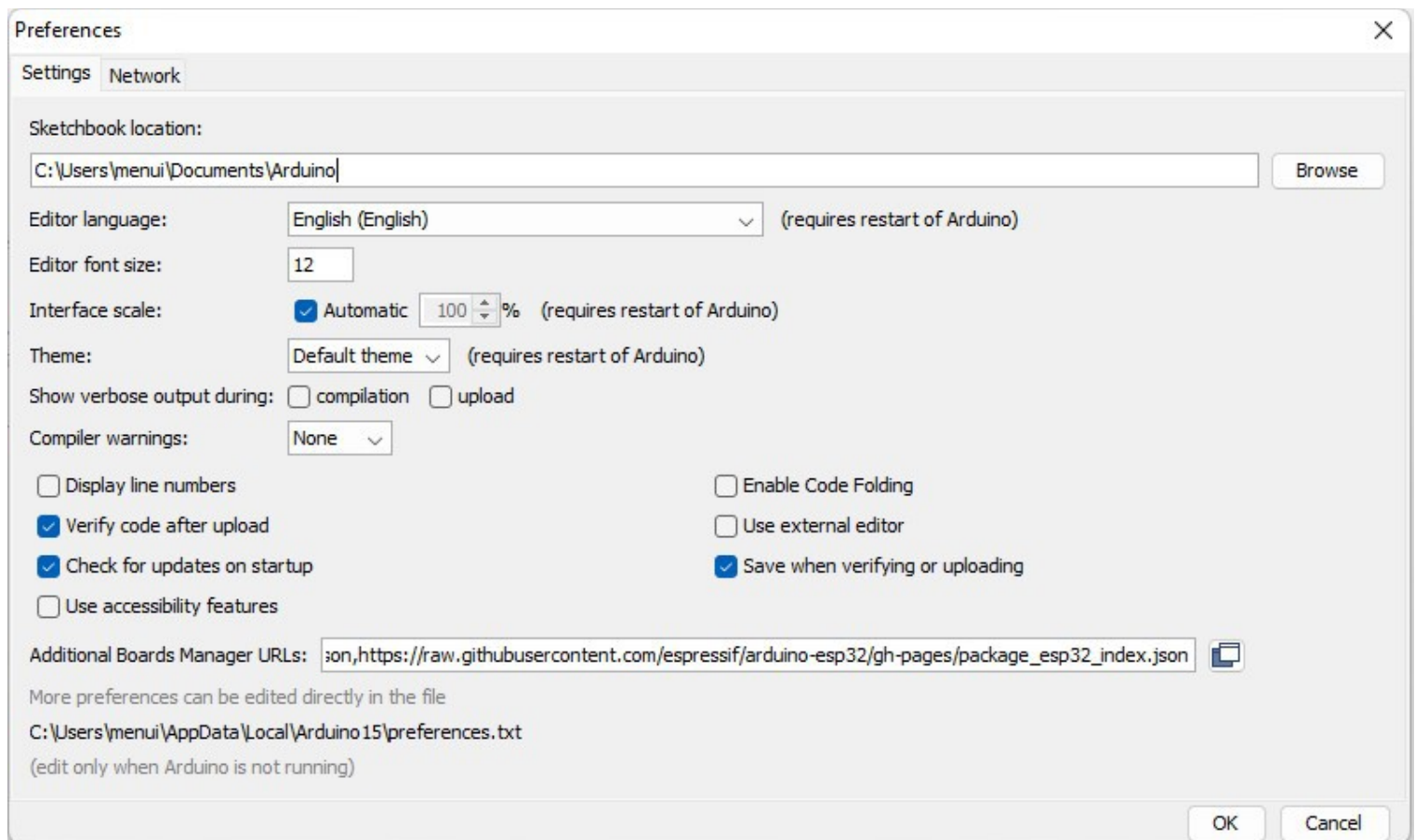Go to www.arduino.cc/en/software and download the software, then install it following the steps.
(You can take IDE Arduino 2.XX or 1.8.X)

2-Add ESP32 board package
Once the installation is complete, launch Arduino IDE, then at the top left go to:
File → Preferences
This window opens:

In the "Additional Map Manager URLs" box, paste this:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

This is the link to files necessary for the installation and use of ESP32 cards, provided by Espressif, the manufacturer.

While in this window you will be able to:
    - modify the language of the software if necessary.
    - note where "Sketchbook location" is, you will need it later.
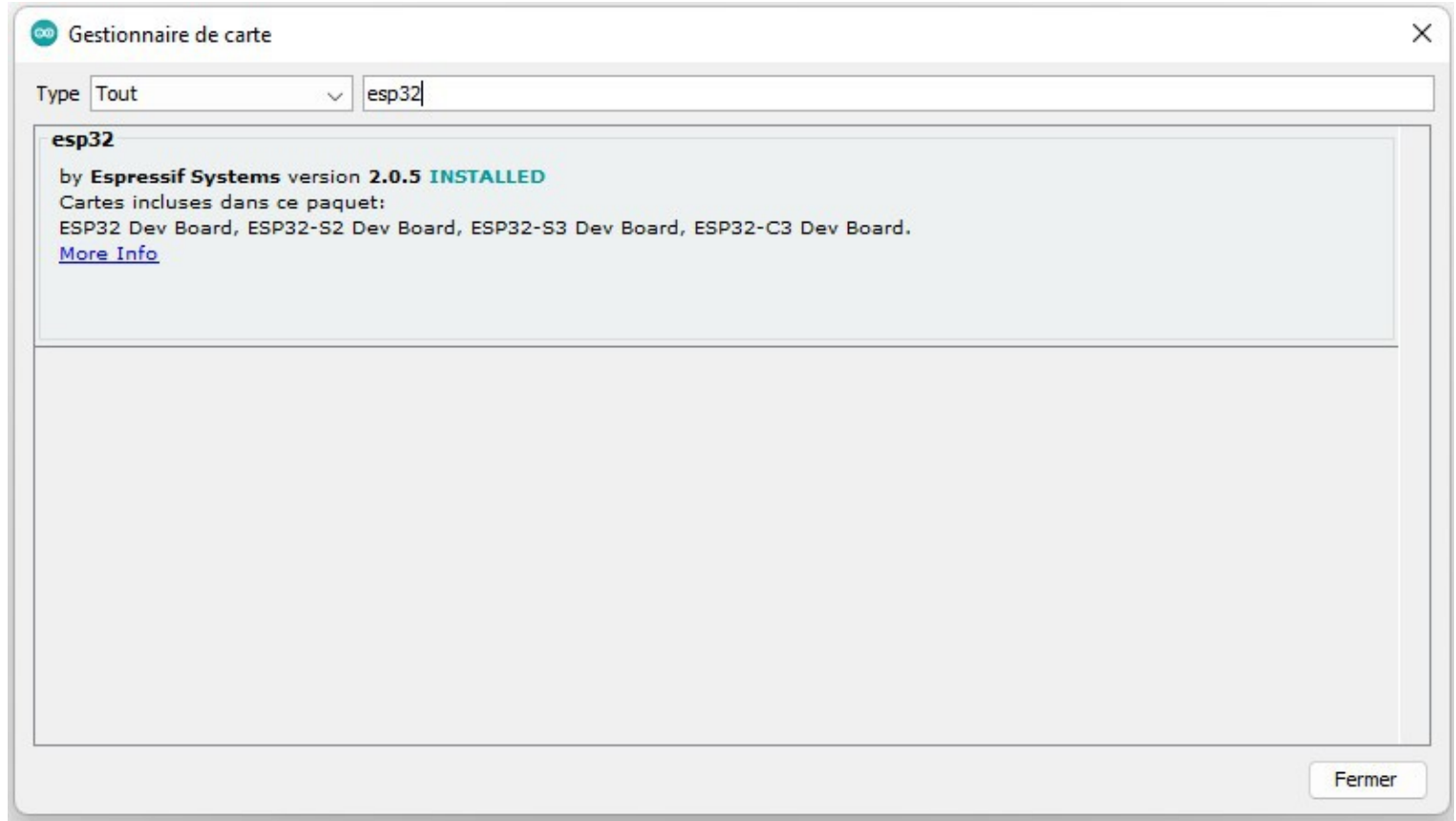
Close the Arduino IDE and relaunch it.

3-Installation the ESP32 board in the IDE
Go to:
Tools → Board → Board Manager
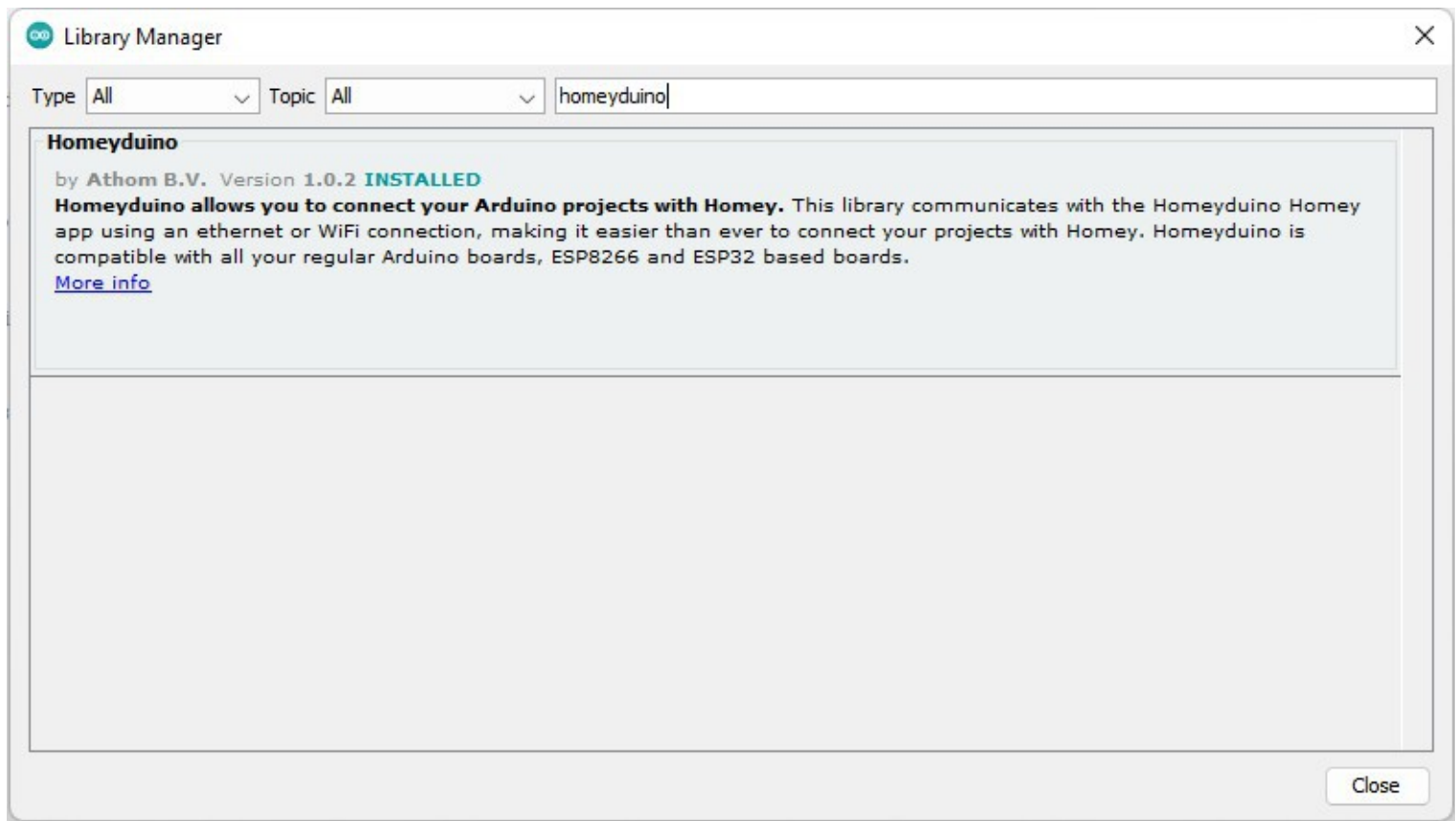
In this window, search for "esp32" and install it.

4-Installation of the necessary libraries

You need to install the "homeyduino" libraries for communication with Homey and "LilyGo-EPD47" for the screen.

Now go to Tools → Manage Libraries

Search and install "homeyduino".
This is a simple way to install a library on Arduino IDE.

For the second library it will be different because it is not referenced in the IDE library manager.

Go to this github page: https://github.com/Xinyuan-LilyGO/LilyGo-EPD47
Click the green button <Code> → Download ZIP.

Unzip this folder into the "libraries" folder located in the "Sketchbook Location" mentioned above.

Normally it is located in: C:\Users\YOUR NAME\Documents\Arduino
(You can find its location in the Arduino IDE File → Preferences)

If the "libraries" folder does not exist, create it and unzip the downloaded "LilyGo-EPD47" folder, in this "libraries" folder

5-Downloading and setting the Homey e-paper screen code

Download my code here: https://github.com/sebyldino/Homey_e-paper

As before: <Code> → Download ZIP

Unzip the folder to the place you want (on your desktop, in your documents...) it doesn't matter.

You should have a "Homey_e_paper..." folder.

In this folder you have several folders and files including a "Homey_e_paper_V1.X.X.ino" (where v1.X.X is the version of course). Make sure this file and the folder it is in have the same name (without the .ino)

You must therefore have for example a "Homey_e_paper_bonjour" folder and in this folder a "Homey_e_paper_bonjour.ino" file, as well as the other folders and files.
(If there is an error, at the next step, a dialog box will tell you that the XXX file must be in the folder of the same name)

Once done, open the Homey_e_paper_XXXX.ino file, you will get here:

```
paperScreen5 §   HomeyAction §   HomeyFunction   ReleaseNote.h   data   debug §   functions §   imagesAndFonts.h §   language   settings.h §   setup §   zoneS
/*================================================================================================
                                        HOMEY e-paper screen
  ================================================================================================*/
//Change your SSID and PASSWORD

const char* ssid     = "YOUR SSID";
const char* password = "YOUR PASSWORD";
|
/*================================================================================================*/



#ifndef BOARD_HAS_PSRAM
#error "Please enable PSRAM in 'Tools > PSRAM!!!'"
#endif
#include <WiFi.h>
#include <WiFiClient.h>
#include <Homey.h>
#include <Arduino.h>
#include <FS.h>
#include <SPI.h>
#include <SD.h>
#include "epd_driver.h"
#include "esp_adc_cal.h"
#include "imagesAndFonts.h"
#include "settings.h"
#include "zoneSettings.h"
#include <Preferences.h>
Preferences preferences;
```

Just replace these 2 lines:

const char* ssid = "YOUR SSID"; ← The name of your router
const char* password = "YOUR PASSWORD"; ← His password

If your box is called "Livebox-1234" and its password is "12345678", you should have:

```
const char* ssid = "Livebox-1234";
const char* password = "12345678";
```

That's all you have to do to the code for it to work. If however you want to change its name go to the next tab "HomeyAction" and change the line:
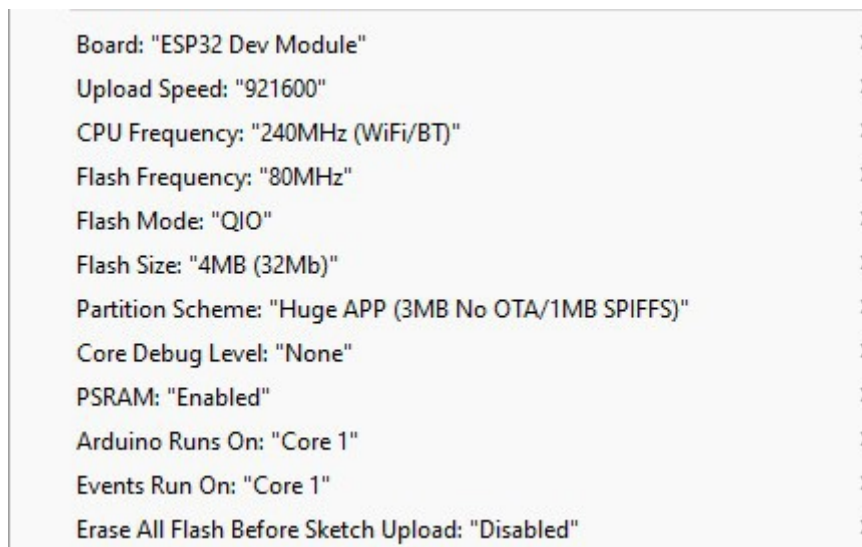
```
Homey.begin("Paper Screen"); //Get name for your device ← Change the name keeping the " "
```

(This is the name that will be displayed by default in Homey, but you can change it from Homey as you change the name of a classic device)
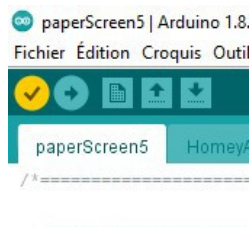
6- Compiling and uploading the code

Now go to Tools → Board Type → Arduino ESP32 and choose "ESP32 Dev Module", The software may show as latency the time to change the card.
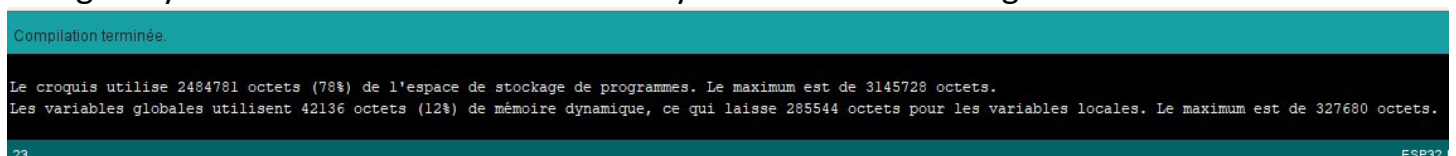
Then go back to Tools and verify that the settings are as follows:

Board: "ESP32 Dev Module"	>
Upload Speed: "921600"	>
CPU Frequency: "240MHz (WiFi/BT)"	>
Flash Frequency: "80MHz"	>
Flash Mode: "QIO"	>
Flash Size: "4MB (32Mb)"	>
Partition Scheme: "Huge APP (3MB No OTA/1MB SPIFFS)"	>
Core Debug Level: "None"	>
PSRAM: "Enabled"	>
Arduino Runs On: "Core 1"	>
Events Run On: "Core 1"	>
Erase All Flash Before Sketch Upload: "Disabled"	>

You can now "Compile" the code by clicking ✓on the top left.

If all is good you should have at the bottom of your screen something like this:

```
Compilation terminée.

Le croquis utilise 2484781 octets (78%) de l'espace de stockage de programmes. Le maximum est de 3145728 octets.
Les variables globales utilisent 42136 octets (12%) de mémoire dynamique, ce qui laisse 285544 octets pour les variables locales. Le maximum est de 327680 octets.

23                                                                                                                    ESP32
```
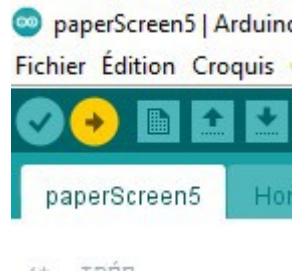
Otherwise the error messages will be displayed in orange, correct them and start again.

When everything is ok, connect your card (your LilyGo) to your computer,
go to Tools → Port and select the port of your card.

Then "Upload" your program by clicking → on the top left.

Wait until it's finished.
It's over for the programming part, the hard part is done! The rest happens on Homey :)

7-Integration in Homey

Download the Homeyduino app here:
https://homey.app/fr-fr/app/com.athom.homeyduino/Homeyduino/

And proceed as for adding a classic device to Homey.
So in Devices → + (top right) → Homeyduino → Arduino → Connection

You should see your device appear. Select it and continue.

Good, your device is connected to Homey!

Thank you, Seby.