



2023

# SCRAPY Coder User Manual

Project number: 2021-1-FR01-KA220-SCH-000031617



 **Co-funded by  
the European Union**

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



## Table of Contents

1. Introduction .....	2
2. Logging in .....	2
3. Home page .....	5
3.1. Language selection .....	6
3.2. Progress .....	7
3.3. All Projects .....	7
3.4. Navigation bar .....	11
4. Blockly programming environment .....	12
5. Badges .....	13
6. Projects solutions .....	14

## 1. Introduction

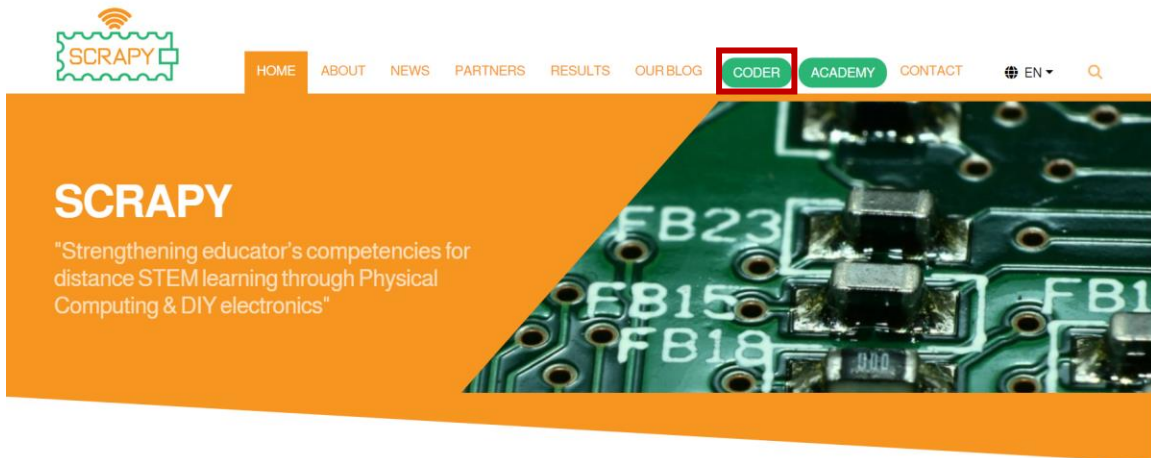
The [SCRAPY Coder](#) is a drag & drop programming application, designed specifically for hosting a series of DIY projects based on the SCRAPY Kit. The programming environment is developed using [Google Blockly](#), with several custom-made blocks of code to serve the requirements of the Kit's electronics, sensors, and components.

Using the Coder, an interested user can program several DIY electronic devices, learning at the same time physical computing concepts, and electricity and circuits paradigms. The Coder also provides the possibility to the user to create their own projects and program them using the coding environment.

This user manual will help users and educators get started with the SCRAPY Coder, understand its various functionalities and features, and finally become familiar with the Blockly programming environment in order to implement the Coder's projects as well their own.

## 2. Logging in

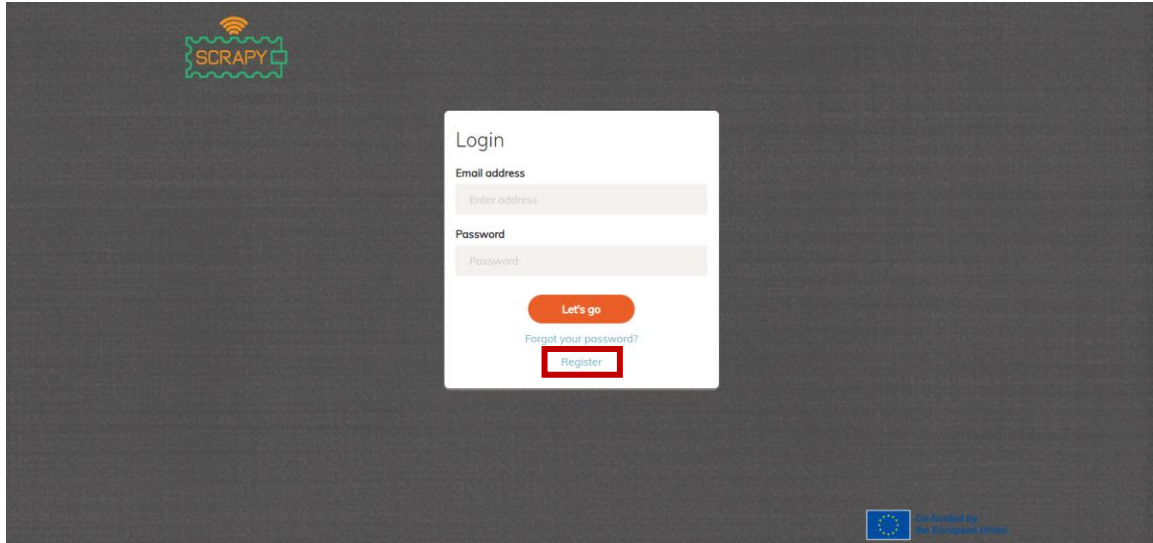
Accessing the SCRAPY Coder, can be done by visiting the project's website ([www.scrapykit.eu](http://www.scrapykit.eu)) and simply click on the "Coder" button on the homepage.



Alternatively, users can use the following url: [www.coder.scrapykit.eu](http://www.coder.scrapykit.eu).



First time users are required to create an account. At the login page, simply click on the “Register” button.



Fill in your information and click on “Register”.

Registration form

First Name

Last Name

Email address

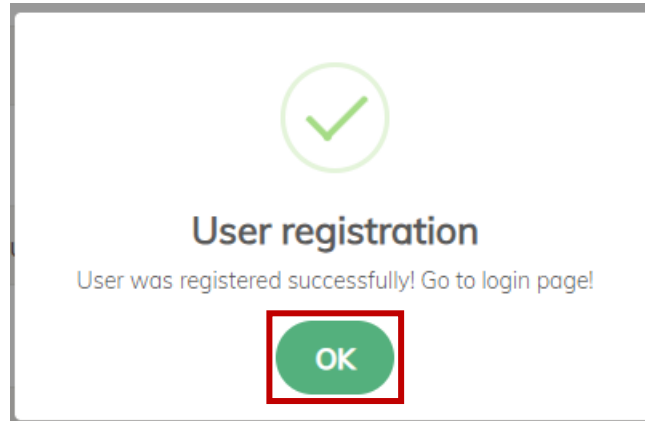
Password

Country

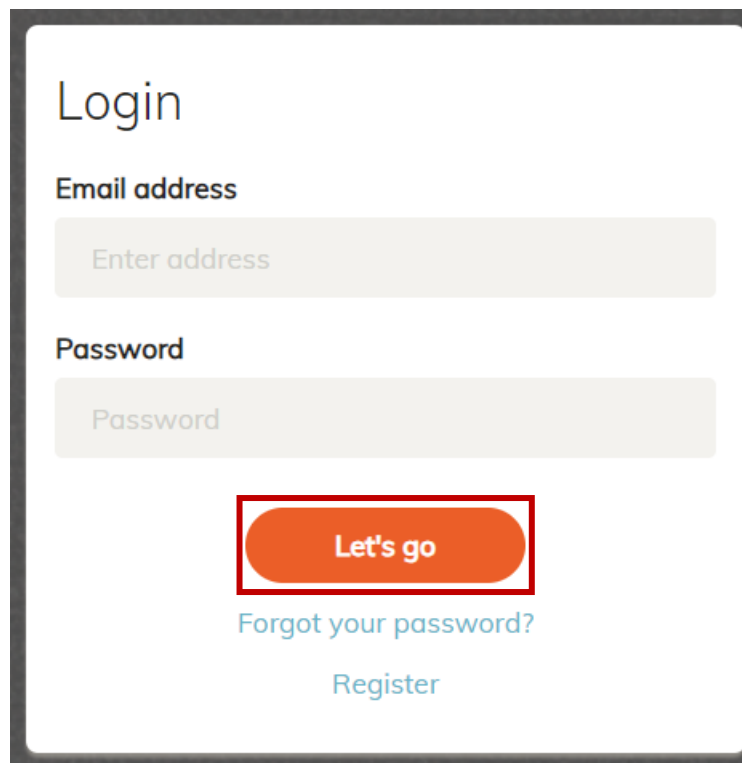
**Register**

[Looking to login?](#)

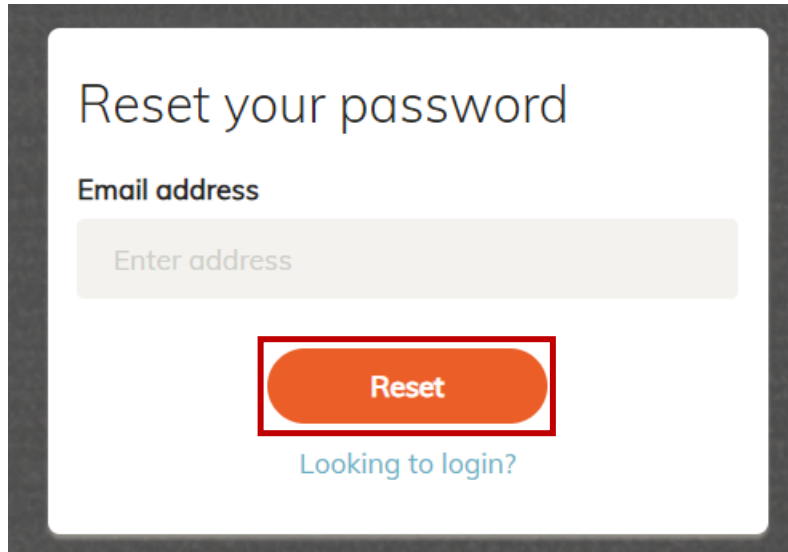
A pop-up message will appear after a successful registration. Click on “OK” and go to the login page by clicking on “Looking to login?”. Additionally, a confirmation email will arrive at your inbox, verifying your registration.



At the login page, insert your email address and password, and click on “Let’s go”.

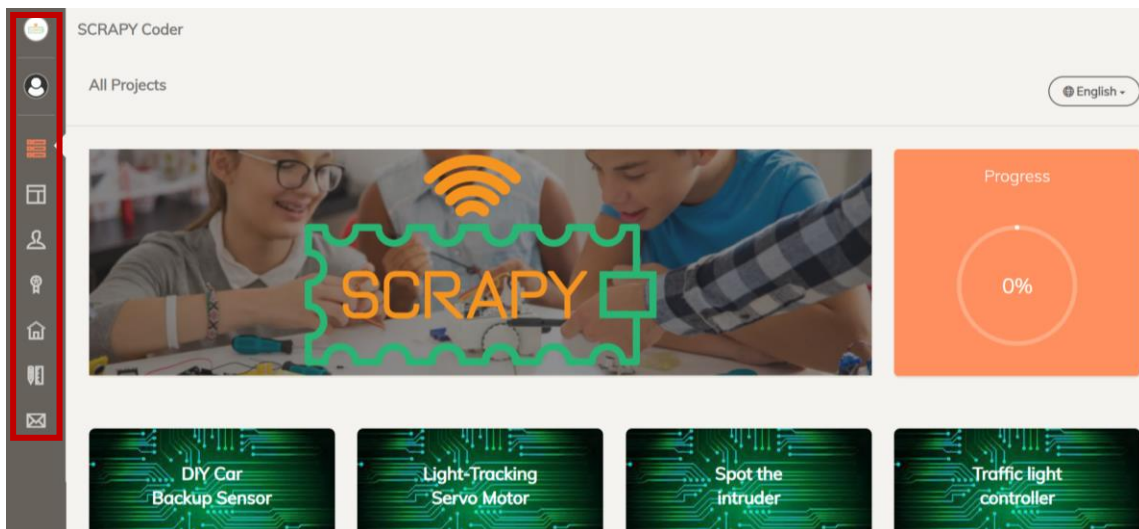
A white rectangular login form with a dark border. At the top, the word "Login" is displayed in a large font. Below it, there are two input fields: "Email address" and "Password", each with a light gray placeholder text. At the bottom, there is a red rounded rectangular button with the text "Let's go" inside, highlighted with a red border. Below the button, there are two links: "Forgot your password?" and "Register", both in blue text.

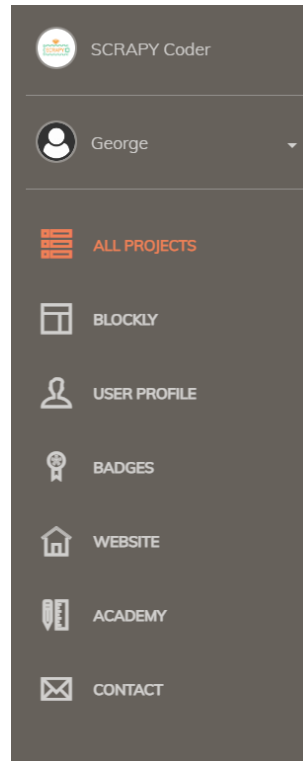
In case you forgot your password, simply click on “Forgot your password?”, and you will be redirected on the password reset page. There, insert your email address and click on “Reset”. An email containing further instructions will arrive at your inbox.



### 3. Home page

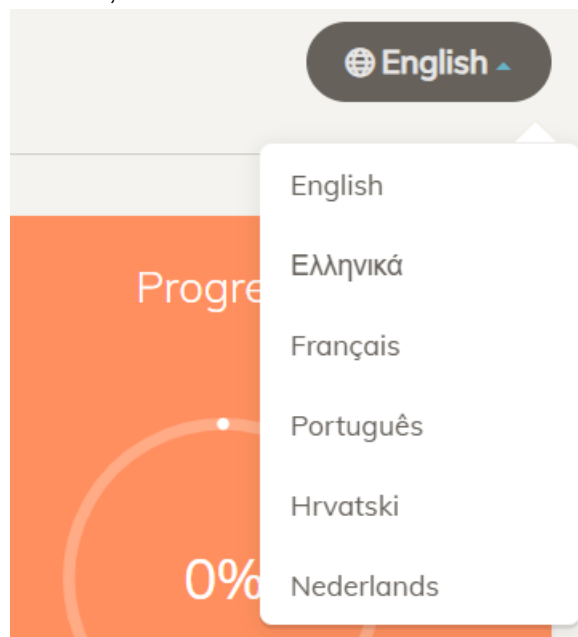
After successfully logging in to your account, you will be redirected to the Coder's homepage. There, the list of available projects will be presented, along with language selection options, progress status and some other information. Most of the Coder's features can be accessed through the navigation panel on the left side of the homepage.





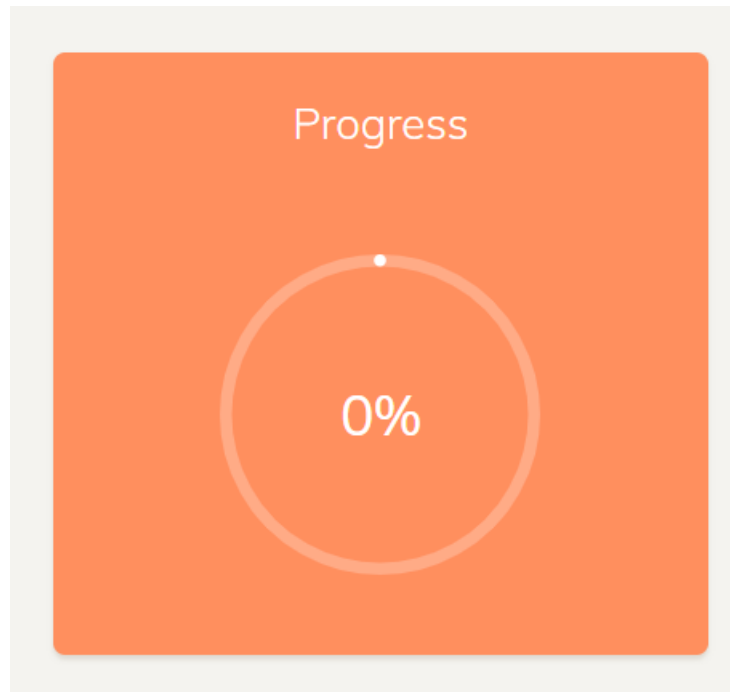
### 3.1. Language selection

On the top right corner, there is a language selector. Simply click on it and choose your preferred language. The SCRAPY Coder is provided in six languages: English, Greek, French, Portuguese, Croatian, and Dutch.



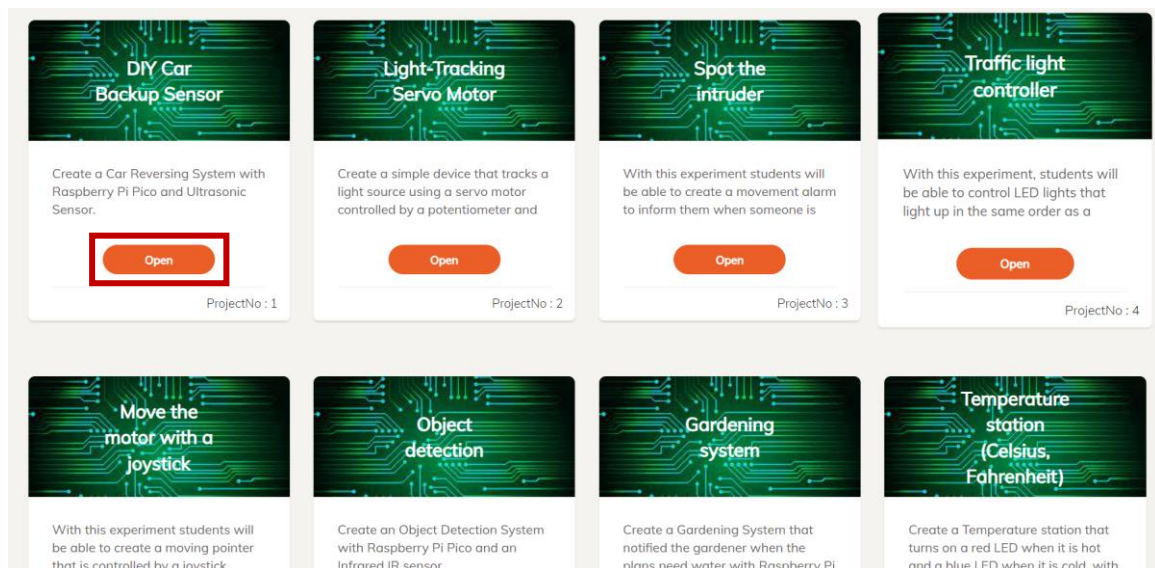
### 3.2. Progress

Directly under the language selector, there is a progress wheel. Each time you complete a project, the percentage will increase until you reach 100% completion.



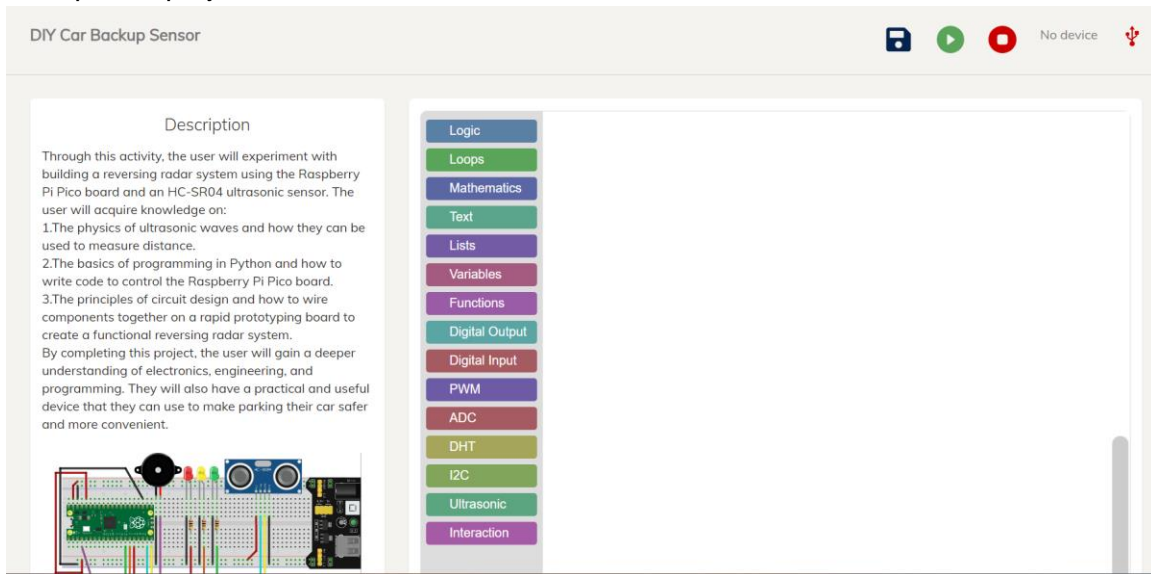
### 3.3. All Projects

The Coder provides access to 12 DIY projects, utilizing the SCRAPY Kit's electronics, sensors and other components. All projects have free access by simply clicking on the "Open" button.

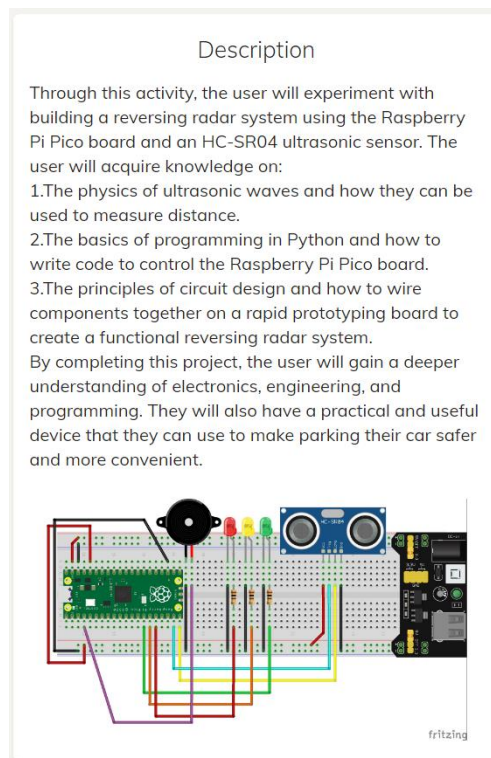




By clicking on the “OPEN” button, the user will be directed to the coding environment for this specific project.



There, the user can see the Blockly coding environment, a description of the project, and a step-by-step guide for implementing the project.



### Step by Step

Connect the HC-SR04 ultrasonic sensor to the Raspberry Pi Pico board using connection wires.

Raspberry Pi Pico Board:

GP15: Trigger pin of the HC-SR04 sensor

GP14: Echo pin of the HC-SR04 sensor

GP10: Positive pin of the green LED

GP11: Positive pin of the orange LED

GP12: Positive pin of the red LED

GP2: Positive pin of the buzzer

GND: Ground pin of the board

HC-SR04 Sensor:

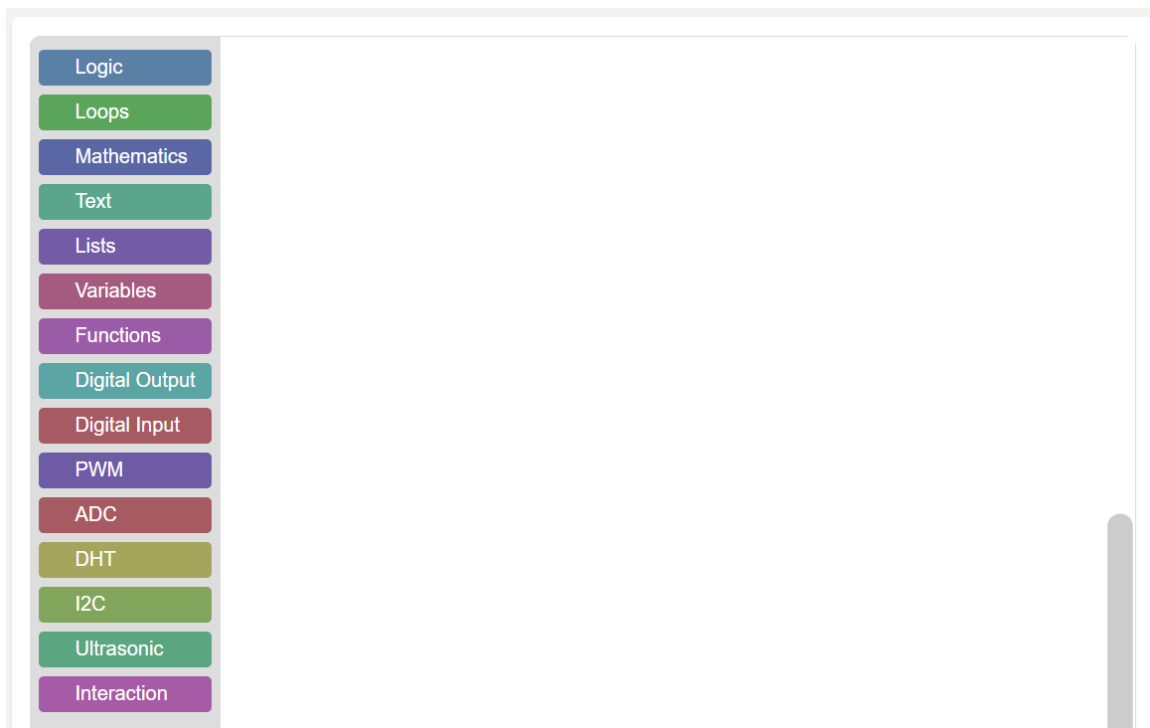
VCC: Connect to 5V power source.

GND: Connect to GND of Raspberry Pi Pico board

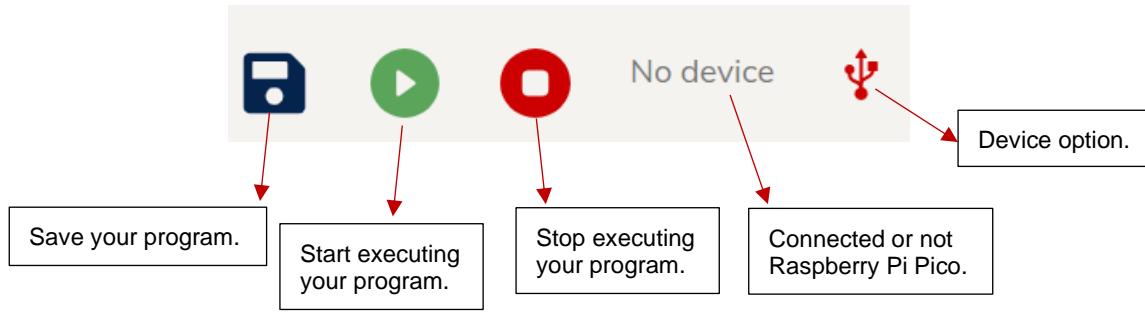
Trig: Connect to GP15 of Raspberry Pi Pico board

Echo: Connect to GP14 of Raspberry Pi Pico board

Next



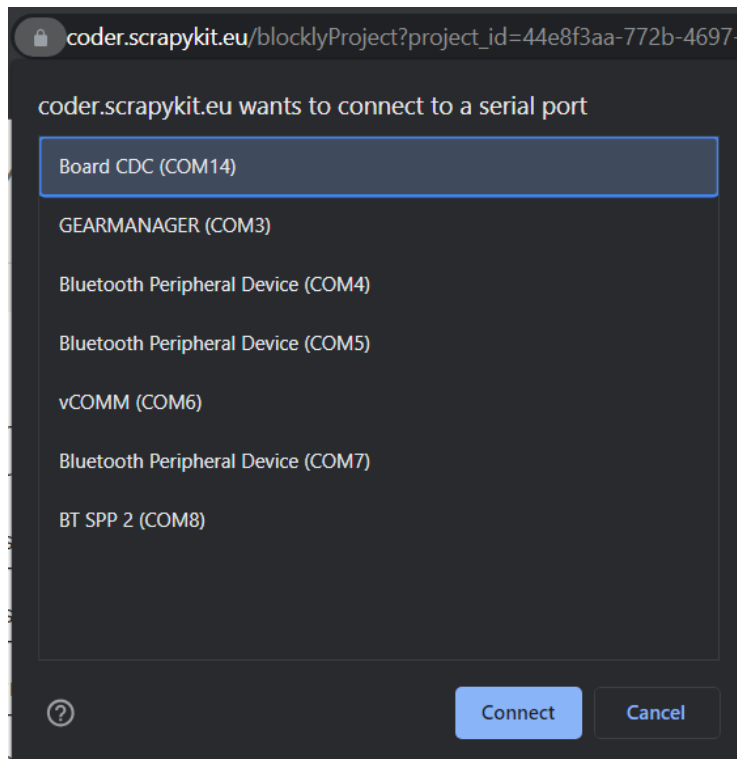
In addition, there are a few options on the top right corner related to the connectivity of the Raspberry Pi Pico.



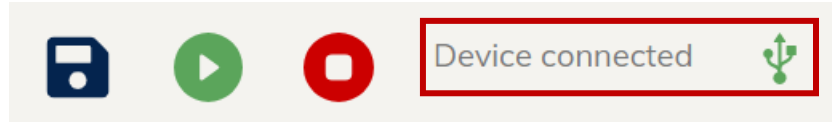
To make the Coder interact with the physical world, you need to connect your Raspberry Pi Pico. The first step is to ensure your Pico has its firmware installed. Browse pages 12-19 of the SCRAPY Kit Manual if you are not sure how to do that. Make sure to close the Thonny Python application. Then, go to the project of your choice, and click on the Device option icon.



A list of all available devices, connected to your computer will pop-up. Choose the one called Board CDC. Then click "Connect".



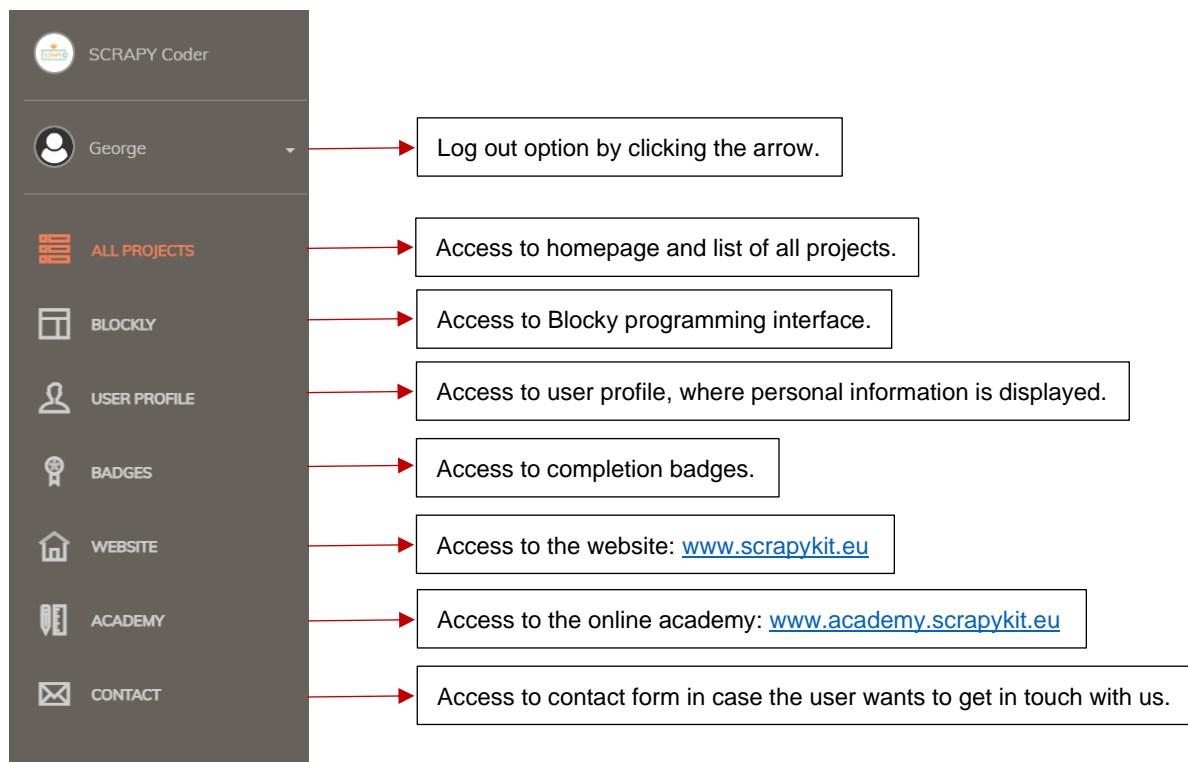
You will get a message that the device is connected, which can also be seen in the top right corner.



Now, you are ready to start building your code for the project of your choice. Once ready, click on the Play button and see what happens in the physical world. You will also get a message that your program is running. If you need to stop your program, simply click on the Stop button.

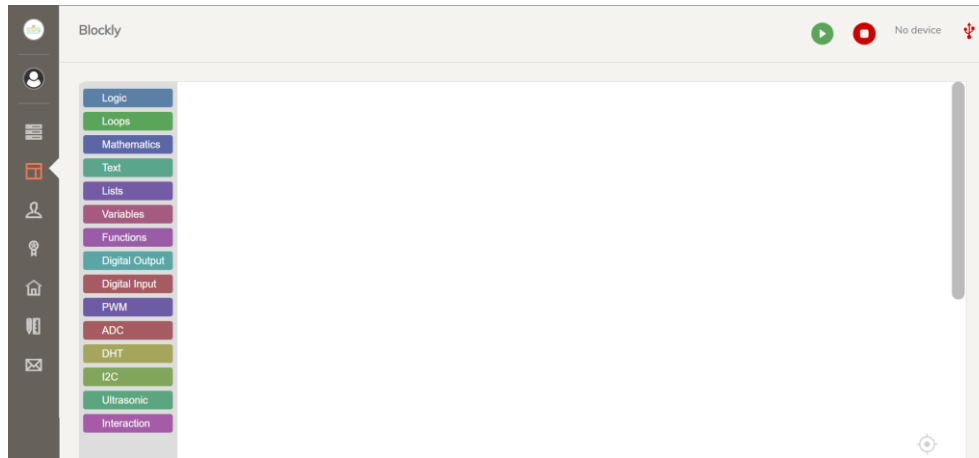
### 3.4. Navigation bar

The navigation bar on the left offers quick access to several features and information related to the Coder, as well as the project SCRAPY.

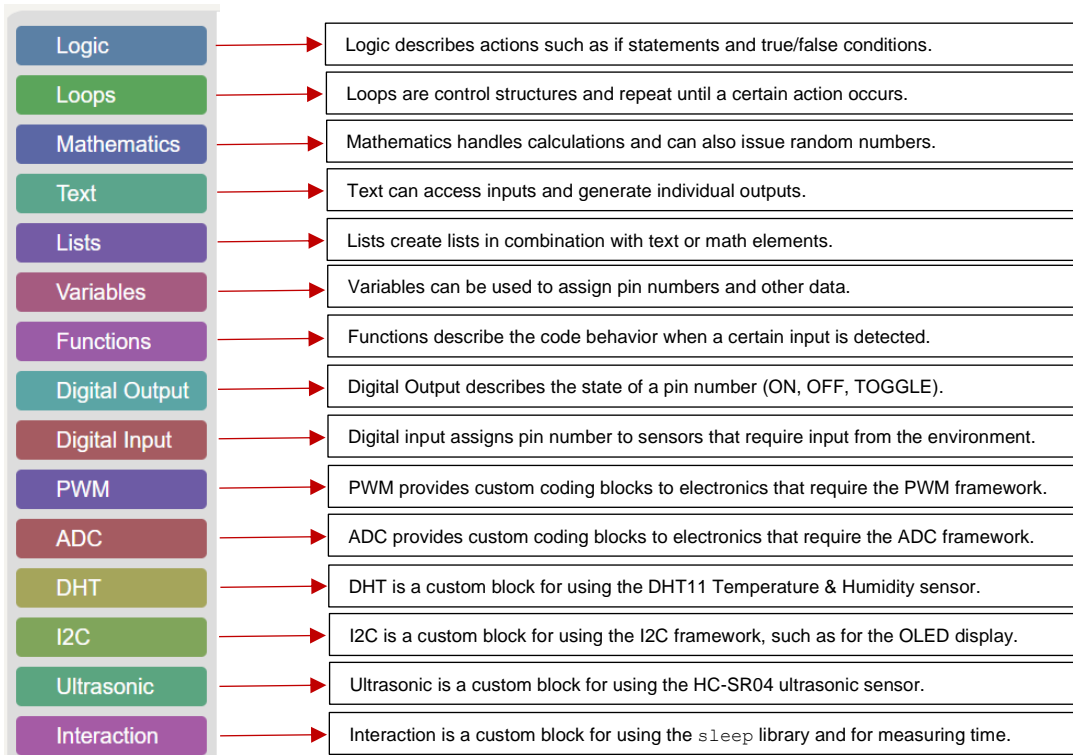


## 4. Blockly programming environment

Accessing the Blockly programming environment can be done by the navigation bar on the left by simply clicking on “BLOCKLY”. The application will redirect you to the following interface:

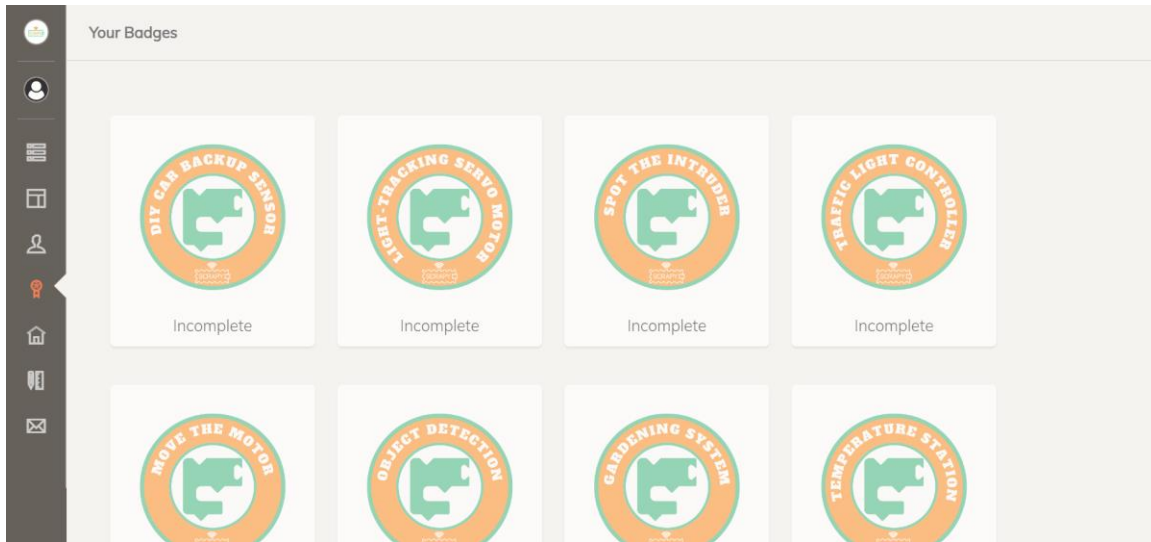


Blockly gives a variety of blocks in order to build different programs depending on your needs. All blocks can be accessed through the “Blocks” bar on the left where all coding categories are displayed. Coding blocks can be dragged and dropped on the white coding space. Each category offers several coding options.



## 5. Badges

SCRAPY Coder awards completion badges for each completed project. These badges can be accessed through the navigation panel on the left, by simply clicking on “BADGES”. In the Badges page, users can see the awarded badges which are marked as “complete”.



## 6. Projects solutions

Project No 1 – DIY car backup sensor

```

set ultrasonic to HC-SR04 ultrasonic sensor with trigger 14 and echo 15
set red_led to Output pin number 12
set yellow_led to Output pin number 11
set green_led to Output pin number 10
set buzzer to Output pin number 2
repeat while true
do
  set distance to Distance in cm ultrasonic
  if distance > 20
  do
    Pin - State ON green_led
    Pin - State OFF red_led
    Pin - State OFF yellow_led
    Pin - State OFF buzzer
  else if distance > 5
  do
    Pin - State ON yellow_led
    Pin - State OFF red_led
    Pin - State OFF green_led
    Pin - State OFF buzzer
  else
    Pin - State ON red_led
    toggle buzzer
    Pin - State OFF yellow_led
    Pin - State OFF red_led
  Sleep 0.1 seconds

```

Project No 2 – Light-tracking servo motor

```

set servo to PWM for pin number 0
set potentiometer_pin to ADC for GPIO26
set ldr_pin to ADC for GPIO27
repeat while true
do
  set potentiometer_value to Read potentiometer_pin
  set ldr_value to Read ldr_pin
  set angle to Angle from potentiometer value potentiometer_value
  set speed to Speed from LDR value ldr_value
  Frequency 50 servo
  set duty to angle ÷ 180
  set duty to round duty ÷ 65025
  Duty_u16 servo
  with cycle duty
  
```

Project No 3 – Spot the intruder

```

set motion_sensor to Input pin number 28 with PULL UP resistance
set buzzer to Output pin number 14
set led to Output pin number 15
repeat while true
do
  if Get value motion_sensor = 1
  do
    toggle buzzer
    toggle led
    Sleep 0.5 seconds
  
```



Project No 4 – Traffic light controller

```

set button to Input pin number 7 with PULL DOWN resistance
set buzzer to Output pin number 16
set green_led to Output pin number 11
set yellow_led to Output pin number 12
set red_led to Output pin number 13
set i2c to I2C 0 for SCL pin 21 and SDA pin 20
set oled to SSD1306_I2C with width 128 and height 64 i2c

repeat while true
do
  Pin - State ON red_led
  Pin - State OFF yellow_led
  Pin - State OFF green_led
  Pin - State OFF buzzer
  Store message PLEASE WAIT in the screen buffer 0 and 10 oled
  Show messages oled
  if Get value button = 1
  do
    Pin - State OFF red_led
    Pin - State ON yellow_led
    Store message CROSSING: in the screen buffer 0 and 20 oled
    Store message ALLOWED in the screen buffer 0 and 30 oled
    Sleep 1.5 seconds
    Pin - State ON green_led
    Sleep 10 seconds
    Show messages oled
    toggle buzzer
    Sleep 0.5 seconds
  
```

Project No 5 – Move the motor with a joystick

```

to map_value with: value, in_min, in_max, out_min, out_max
set value_in_min to value - in_min
set out_max_out_min to out_max - out_min
set value_in_min_out_max_out_min to value_in_min * out_max_out_min
set in_max_in_min to in_max - in_min
set in_max_in_min_out_min to in_max_in_min + out_min
set map_value_return to value_in_min_out_max_out_min ÷ in_max_in_min_out_min
return round map_value_return

set x_axis_pin to 26
set servo_pin to 13
set servo_max_angle to 45
set servo_min_angle to 0
set x_axis_adc to ADC for pin GPIO26
set servo_pwm to PWM for pin number 13
Frequency 30 servo_pwm
Duty_u16 with cycle 0 servo_pwm
repeat while true
do
set x_axis_val to Read x_axis_adc
set angle_x to map_value with:
value x_axis_val
in_min 0
in_max 65535
out_min servo_min_angle
out_max servo_max_angle
Duty_u16 servo_pwm
with cycle map_value with:
value angle_x
in_min servo_min_angle
in_max servo_max_angle
out_min 50
out_max 5000
Sleep 0.5 seconds
  
```

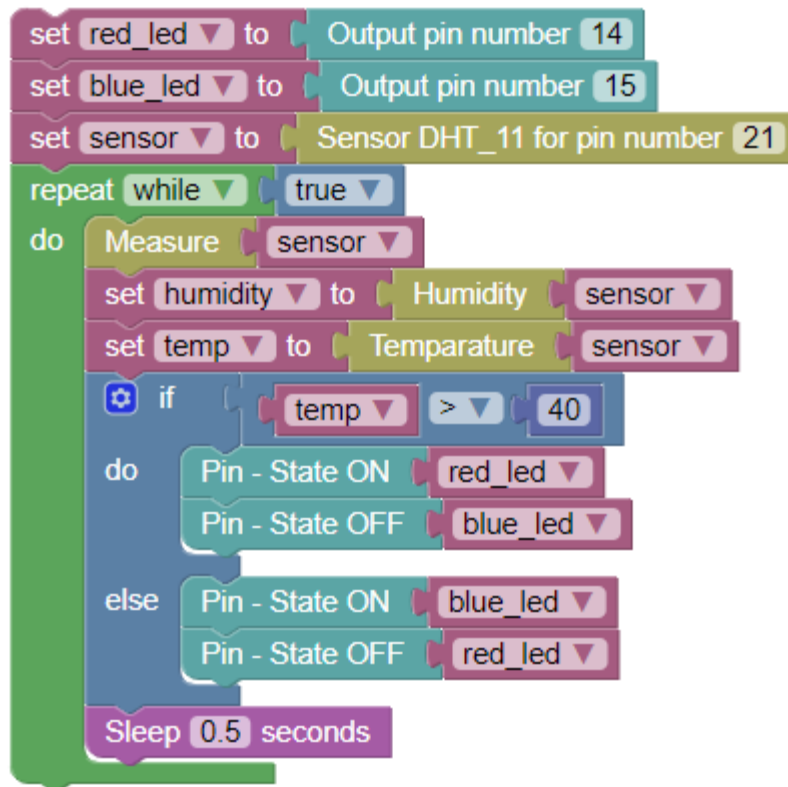
## Project No 6 – Object detection

```
set ir_pin to Input pin number 7
repeat while true
do
  set ir_state to Get value ir_pin
  Sleep 0.5 seconds
```

## Project No 7 – Gardening system

```
set led_pin to Output pin number 15
set sensor pin to ADC for pin GPIO26
set threshlod to 40000
repeat while true
do
  set humidity to Read sensor pin
  if humidity > threshlod
  do
    Pin - State ON led_pin
  else
    Pin - State OFF led_pin
  Sleep 0.5 seconds
```

## Project No 8 – Temperature station (Celsius, Fahrenheit)



Project No 9 – Fire alarm

```

set green_led to Output pin number 5
set red_led to Output pin number 4
set buzzer to Output pin number 17
set flame_sensor to Input pin number 28
repeat while true
do
  if Get value flame_sensor = 1
  do
    Pin - State ON red_led
    Pin - State ON buzzer
    Pin - State OFF green_led
    Sleep 1 seconds
  else
    Pin - State ON green_led
    Pin - State OFF red_led
    Pin - State OFF buzzer
    Sleep 1 seconds

```

Project No 10 – Smart clothes dryer

```

set buzzer to Output pin number 17
set rain_sensor to Input pin number 1
repeat while true
do
  if Get value rain_sensor = 0
  do
    Pin - State ON buzzer
    Sleep 1 seconds
  else if Get value rain_sensor = 1
  do
    Pin - State OFF buzzer
    Sleep 1 seconds

```

Project No 11 – Knocking light

```

set red_led to Output pin number 18
set vibration_sensor to Input pin number 21
repeat while true
do
  if Get value vibration_sensor = 0
  do
    Pin - State ON red_led
    Sleep 5 seconds
  else if Get value vibration_sensor = 1
  do
    Pin - State OFF red_led
    Sleep 5 seconds

```

Project No 12 – Sound detection warning alarm

```

set buzzer to Output pin number 18
set sound_sensor to Input pin number 21
repeat while true
do
  if Get value sound_sensor = 0
  do
    toggle buzzer
    Sleep 5 seconds
  else if Get value sound_sensor = 1
  do
    Pin - State OFF buzzer
    Sleep 5 seconds

```