# Red Pitaya Visual Programming Setup

I set up my Red Pitaya for Visual Programming before attempting to connect to it with GNU Radio and Vivado. Visual programming is a web-based approach to programming that simplifies coding down to drag-and-drop blocks. It's intended to make Red Pitaya more accessible in educational situations.

In order to use Red Pitaya visual programming, there is a license that you have to buy. At the time of this writing, the cost was 5 euros per month. You subscribe through the Red Pitaya store. I did this and it was fast and easy, just like all of the other Red Pitaya store experiences have been so far.

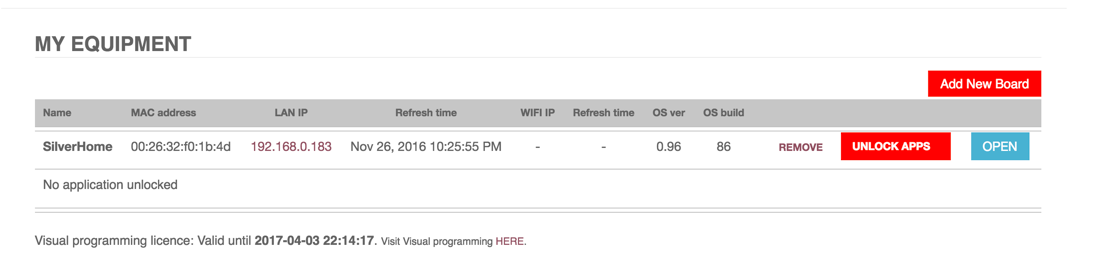
Visual Programming on the Red Pitaya requires internet access. Internet access is also required when upgrading the Red Pitaya operating system, installing applications from the marketplace, and unlocking licenses for applications from the marketplace.

Traditional coding on the Red Pitaya, like using itwith GNU Radio and/or Vivado, does not require internet access.

Basic setup for the Red Pitaya needs to be accomplished before setting up Visual Programming.

The major tasks are unboxing, obtaining a proper power supply, and being able to connect to it by using a web browser. Documentation on how to accomplish the above setup can be found at:  
<http://redpitaya.readthedocs.io/en/latest/doc/quickStart/needs.html>You will need to log in to your Red Pitaya account. You will then add your board to your account. When successful, the board should show up in the "my equipment" page. For me, the link was   
<https://store.redpitaya.com/myequipment/list/>

And the page (after setting things up) looked like this:



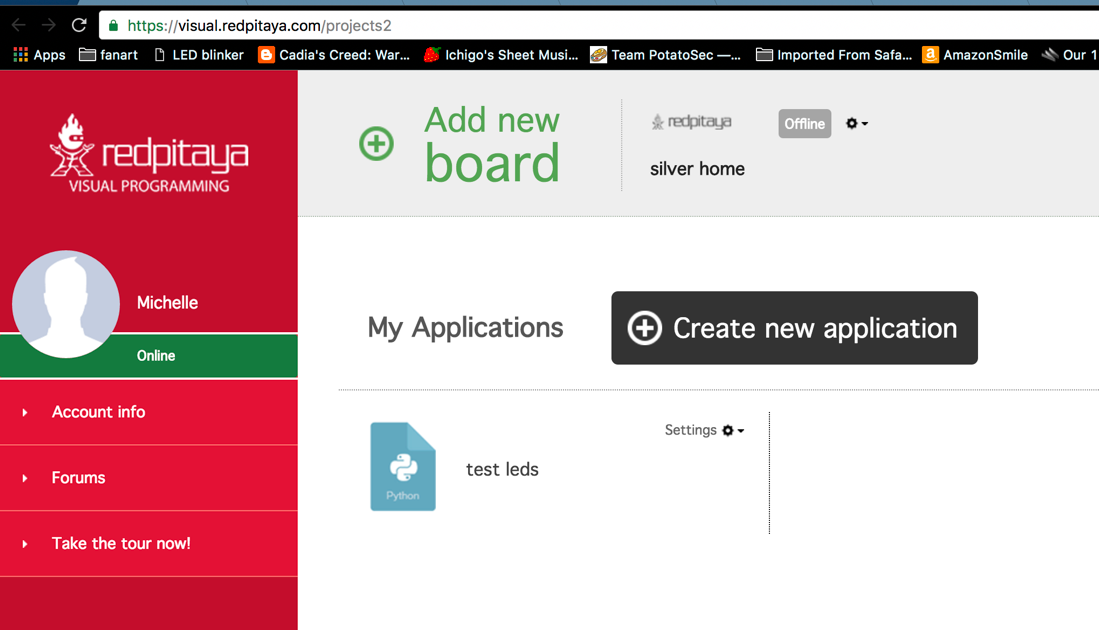
Setting up Visual Programming can be a bit tricky. It's not hard, but there are some fiddly bits.

Click the link Visit Visual programming [HERE](https://visual.redpitaya.com/).

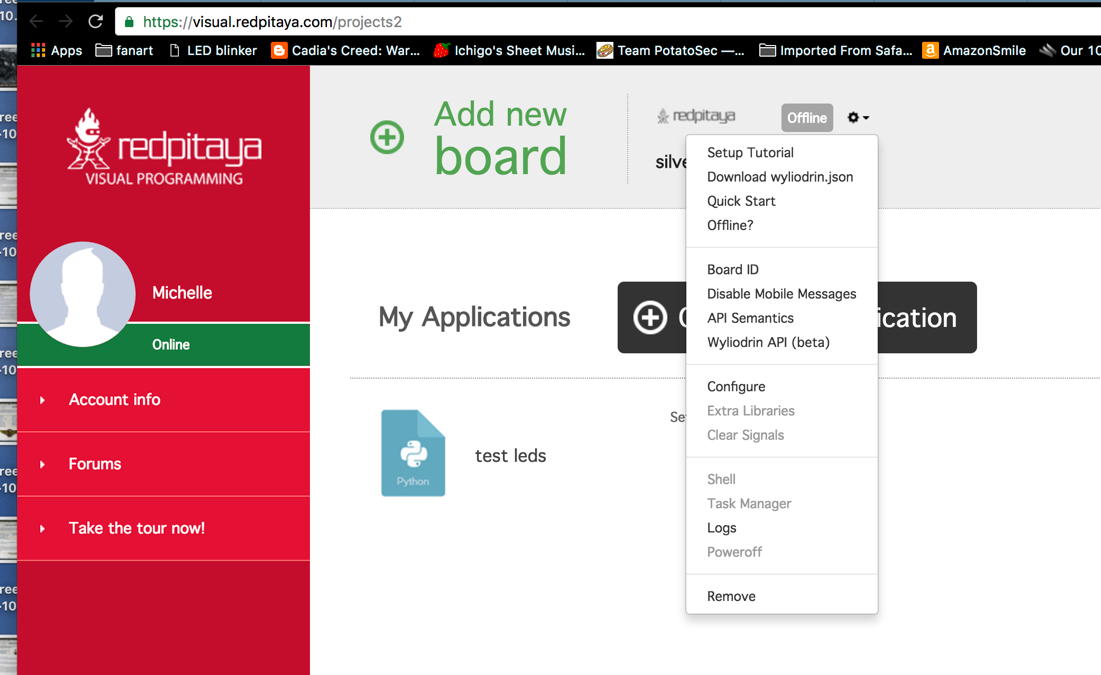
Launches the page

<https://visual.redpitaya.com/>

You should see a page that looks something like this. You'll need to add your board.



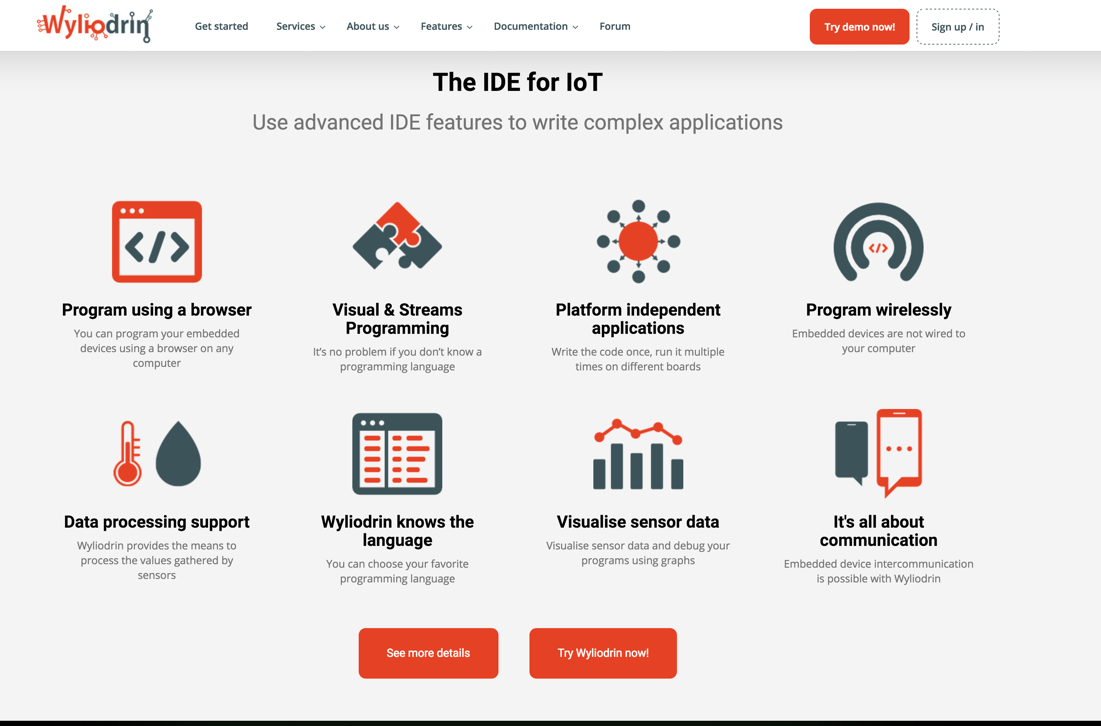
There is a gear menu at the top that we need to use for several important functions.



After you add your board, you are going to have to configure, get extra libraries, and download the wyliodrin.json file. This JSON file must be put on your SD card.

What is wyliodrin? It's an integrated development environment intended to serve the Internet of Things marketplace.

<https://www.wyliodrin.com/>



So, it looks like Red Pitaya is using the wyliodrin IDE to provide the Visual Programming platform.

Interestingly, Red Pitaya is not listed on the wyliodrin site as one of the boards the downloaded version of the IDE supports. However, they do have a video tutorial that shows wyliodrin visual programming in action.

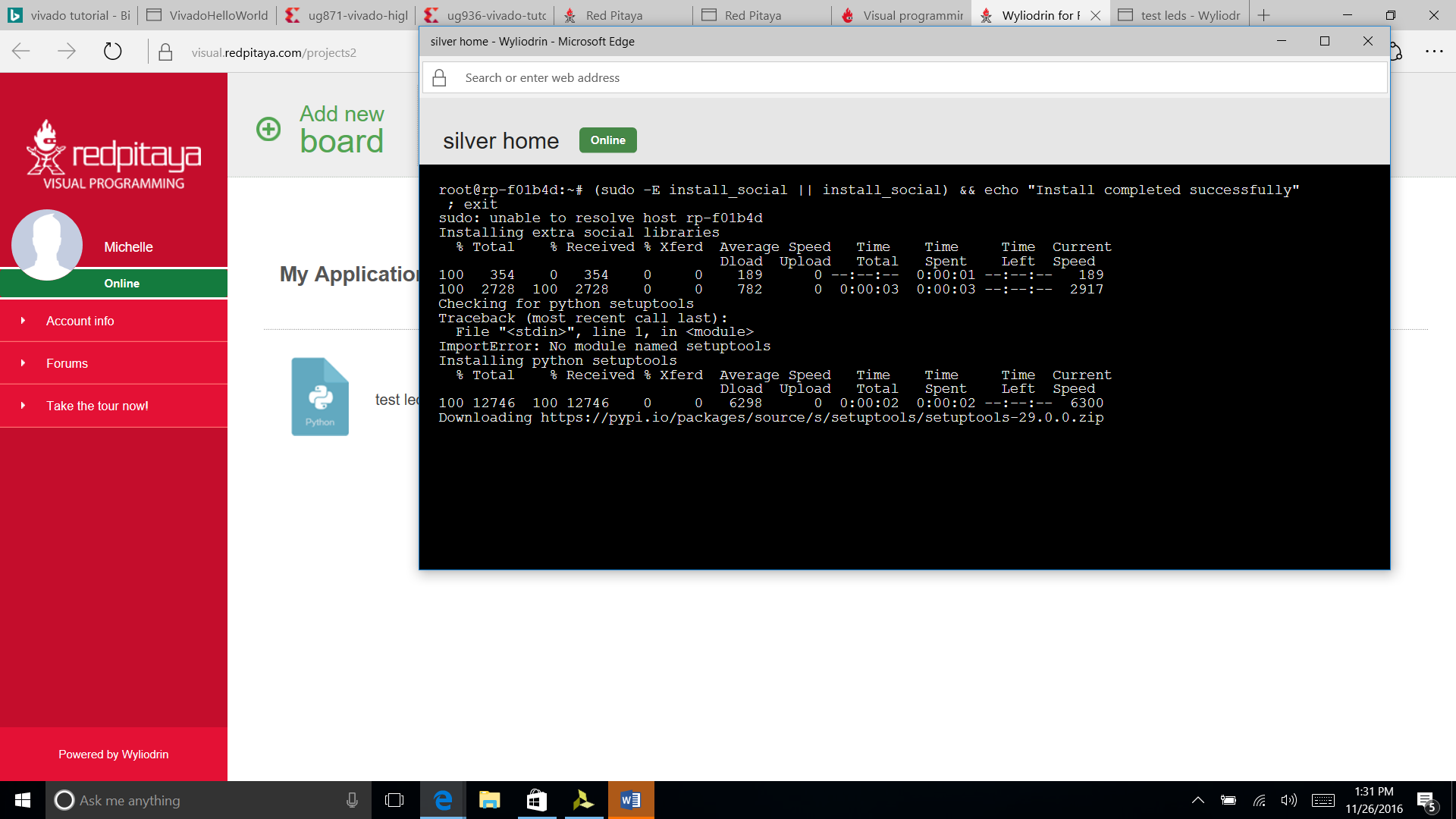
<https://projects.wyliodrin.com/wiki/video_tutorials/visual>

The example video tutorial is a radio project on a Raspberry Pi. It's a radio with a VU meter. Not a bad "hello world" project. In wyliodrin, the blocks (think of them as Legos) that form the visual programming are compiled into Python code. There's a pane for viewing this code within the IDE.

JSON stands for JavaScript Object Notation. JSON files are lists of attributes and value pairs. The lists can be gathered up into JSON objects and/or expressed as JSON arrays. Arrays and objects can be nested.

JSON files are fetched and key-value data extracted. JSON objects are used to create very powerful connected experiences. It's an open standard. Almost all of the applications for the Red Pitaya are enabled by JSON. Your web browser is the front-end, and your Red Pitaya is the back-end. Data between the Red Pitaya and the web browser is passed back and forth in JSON format. This is essentially the same strategy as Phase 4 Ground radios will use.

Here's what it looks like when you install extra libraries.

  
  
The Setup Tutorial link in the gear menu has all the steps we've discussed so far and a direct link to the JSON file.

Without this JSON file, you can click on the visual programming link all day long and nothing productive will happen, but it won't give you a useful error message. Ask me how I know.

In the process of handling the JSON file on a Windows machine, I found out about brackets, which is a "modern open source text editor that understands web design". I found it useful. It can be found here:

<http://brackets.io/>

The JSON file is human readable and as of mid-December 2016 was (for me) looked something like this.

{

"jid": "[abraxas3d\_silver\_home@visual.redpitaya.com](mailto:abraxas3d_silver_home@visual.redpitaya.com)",

"password": "<REMOVED>",

"socketpassword": "<ALSO\_REMOVED>",

"owner": "[abraxas3d@visual.redpitaya.com](mailto:abraxas3d@visual.redpitaya.com)",

"timeout": 2000,

"maxBuffer": 200,

"firewall": false,

"ping": 50,

"ssid": "",

"scan\_ssid": 1,

"psk": ""

}

This file controls relationships between your Red Pitaya board and several other entities.

The board will send status message to Red Pitaya servers for "improvement purposes". This can be turned off by adding a line to the wliodrin.json file.

Add

"privacy":true

Somewhere within the curly brackets.

If only online privacy were always this easy everwhere.

What do the other lines of this JSON file do?

jid and password are XMPP connection credentials. Wyliodrin servers use the listrophe library. The jid is used to send messages to the Red Pitaya. The board ID can be found in the gear menu option.

The other pairs of values seem to support common server-client communications functions. If I find out any additional functions that can be enabled through this file, I'll include them in a future revision. If you know of any, then please feel free to update this file or send me edits.

# Using Visual Programming

Here's a simple LED blinking experiment within Visual Programming for the Red Pitaya.

1. Repeat block – Will cause continuous executions of everything which is inside the block, i.e. while loop.   
2. Inside the Repeat block we have put two Set Led blocks for switching ON and OFF the LED.   
3. Between the ON and OFF states we have added some time delay so we can follow LEDs blinking.

What are the advantages of using Visual Programming?

Ability to create own dashboards with real time graphs, dials, meters, sliders, and buttons • Ability to control the program flow from a PC, smartphone or tablet • Ability to share measurements or send notifications to email or even social networks like Facebook and Twitter • Measures temperature, moisture, alcohol, water level, vibrations, UV light, sound, pressure, air quality detect motion, and other • Controls actuators and indicators like LEDs, displays, motors or relays in order to control high load devices\*The last two features require the use of the Red Pitaya Sensor extension module & sensors • Programming with blocks is a very fun experience, but is also highly instructive and encourages the user to begin thinking subconsciously like a real programmer. All of this is just the beginning of the learning process. This format also enables users to watch and learn what the real programming language code behind the graphical blocks looks like – and how to program using it.

7