# GranaSat Batteries Tester

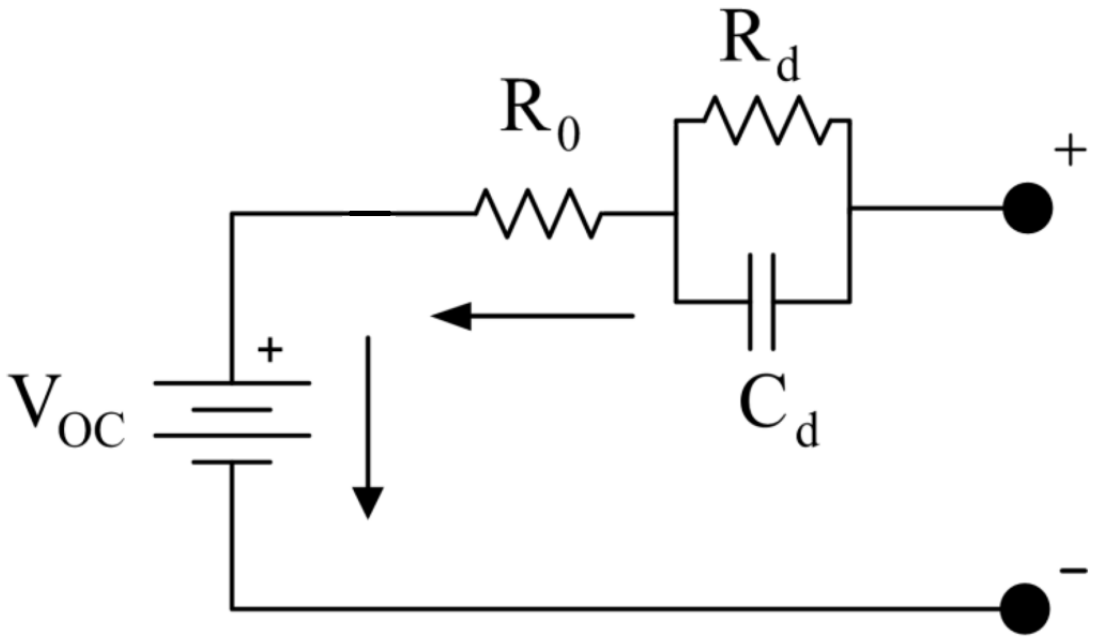
Blanca Cano Camarero

Miguel Romero García

# Index

# Planning

We will test batteries in some different situations. First, in the picture we can see the lithium ion battery model, which has a voltage source that charge the battery, and at the end, the points where you connect the load in order to discharge the battery




**SOFTWARE**

**INTERFACE**

Blanca will made the interface of the program for control the batteries and the different tests.

**TEST**

Miguel will made the code for control the differents equipments to use in the test

**HARDWARE**

**TEST**

We will made the differents test using the equipment like the vacuum chamber

## Test #1: High Vacuum Test


- Battery capacity
- Battery loss percentage after vacuum
- Change in size after vacuum test

## Test #2: Capacity and Internal Resistance vs Temperature

- Battery capacity versus temperature
- Battery internal resistance versus temperature
- Charging/ discharging voltage and current curves at different temperatures.
- Battery size in charged and uncharged state

C → Measure of the rate at which a battery is discharged relative to its maximum capaity

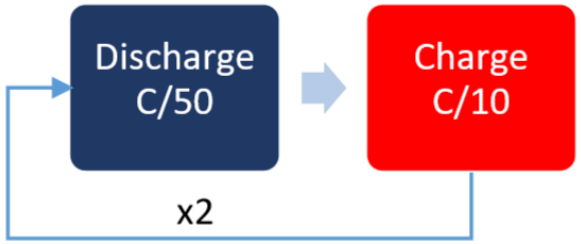
## Test #3: Self-Discharge Test


- Battery loss of energy over long periods.

## Test #4: LEO Cycling

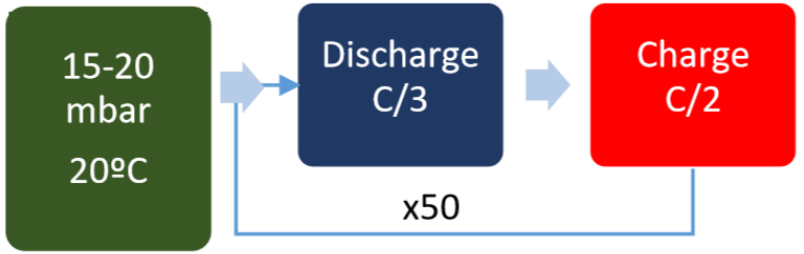

- Battery degradation over continuous charge/discharge periods at border temperatures.
- Battery charge/ discharge curve degradation.

## Test #5: EMF vs SOC




- Battery curve of the electromotive force versus the state of charge

## Test #6-7: Reduced pressure 30 %/80% DOD Cycling




- Evolution of EODV (End of Discharge Voltage) versus degradation.
- Evolution of the battery capacity and efficiency versus degradation

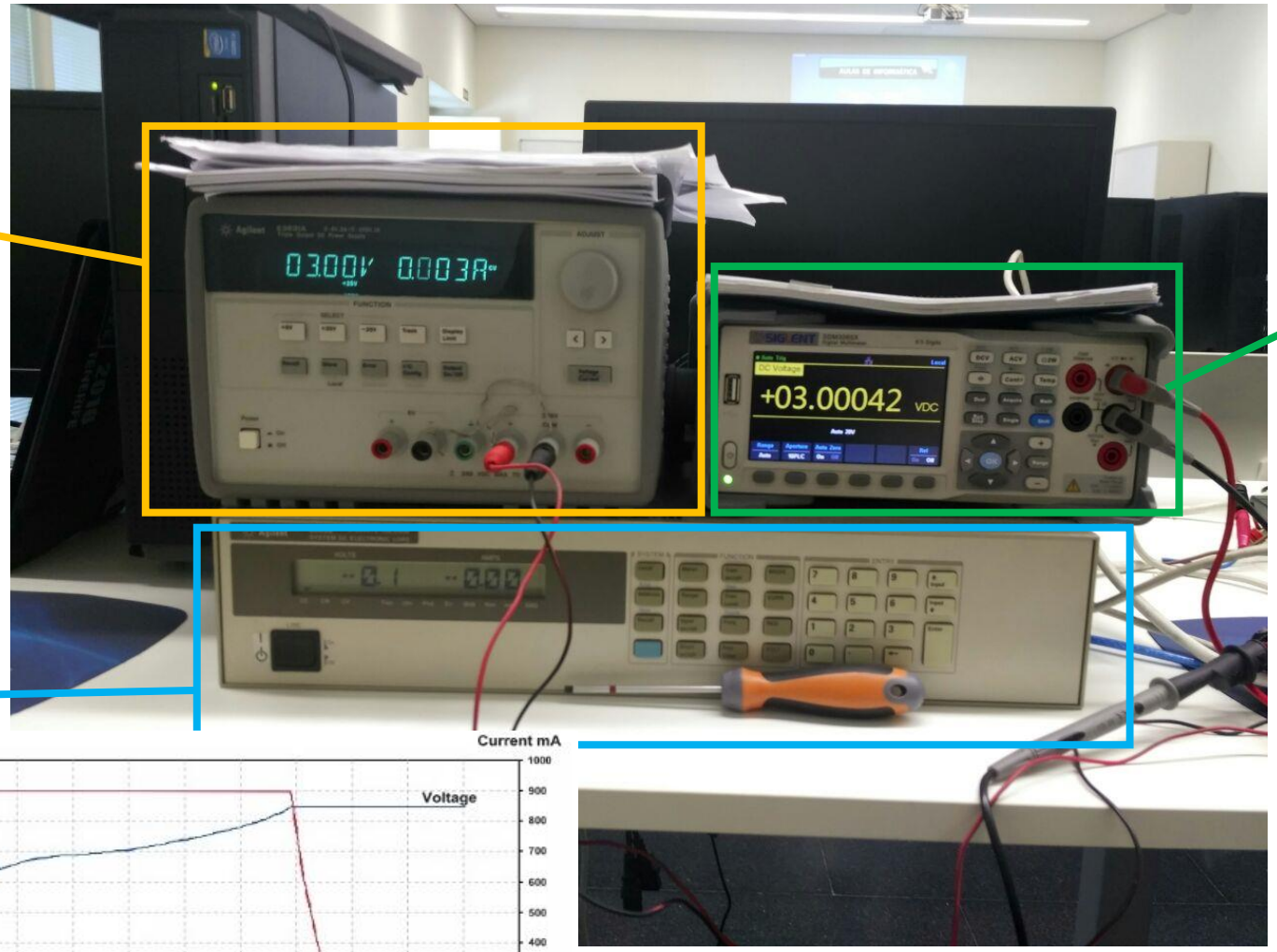
# Equiptment Used

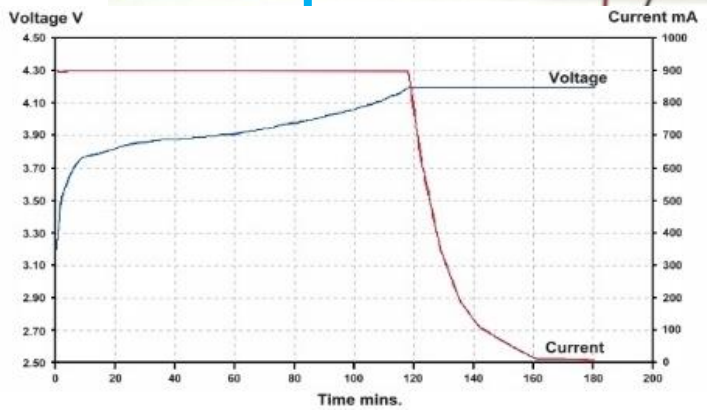
## E3631A DC-Supply

The triple output supply offers three independent outputs: 0 to 6 V/5A, 0 to +25V/1A and 0 to –25V/1A. The 6 V output is electrically isolated from the ±25 V supply to minimize any interference between circuits under test. The ±25 V outputs can be set to track each other.

## 6063B DC-Load

- Current: 0 to 10 A
- Volts: 3 to 240 V
- Maximum power: 250 W



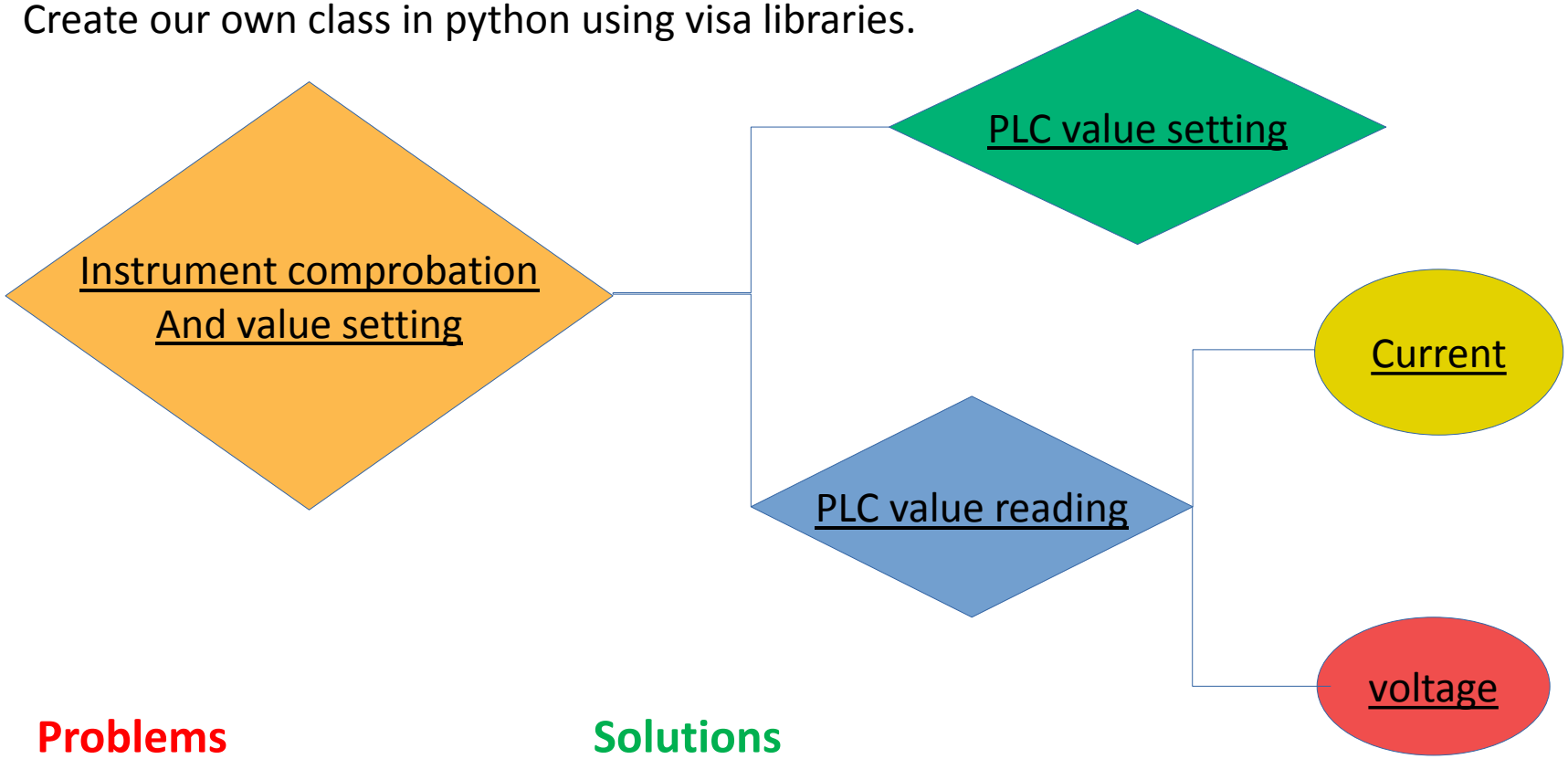




## Multimeter SDM3065X

The multimeter read the voltage and current of the battery, and we use it because the 6063B DC-Load hasn't got the same precision as the multimeter

# Multimeter control, software

Create our own class in python using visa libraries.





**PLC**

- Average in time of
- Current
- Capacity
- Work at 60hz

**Problems**

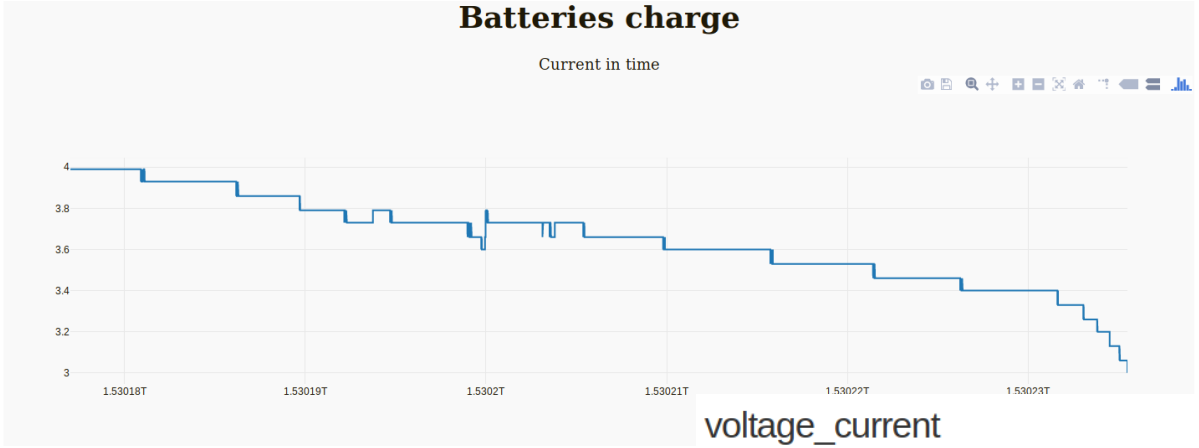
MEAS::CURR
Reconfigure all instrument

**Solutions**

Create our own funtion of measure.

```
def current(self):
        self.instr.write('CONF:CURR:DC')
        self.instr.write(f'CURR:DC:NPLC {self.plc_current}')
        return float(self.instr.query('READ?'))
```

# Interface evolution

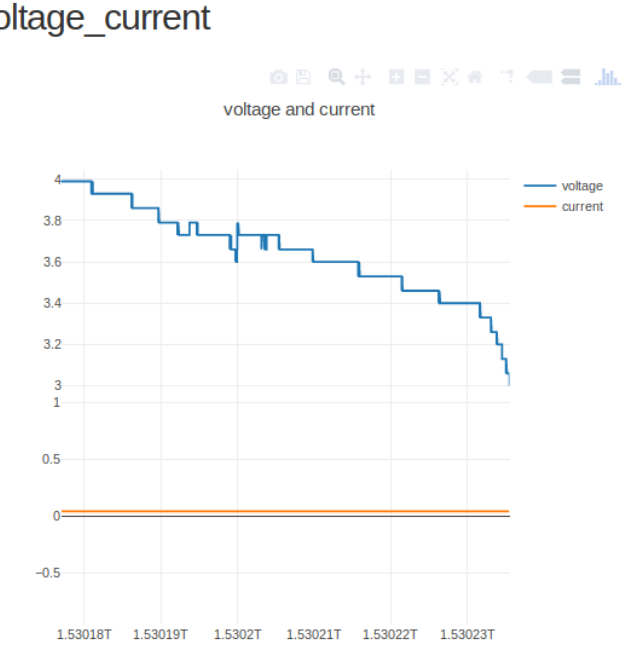


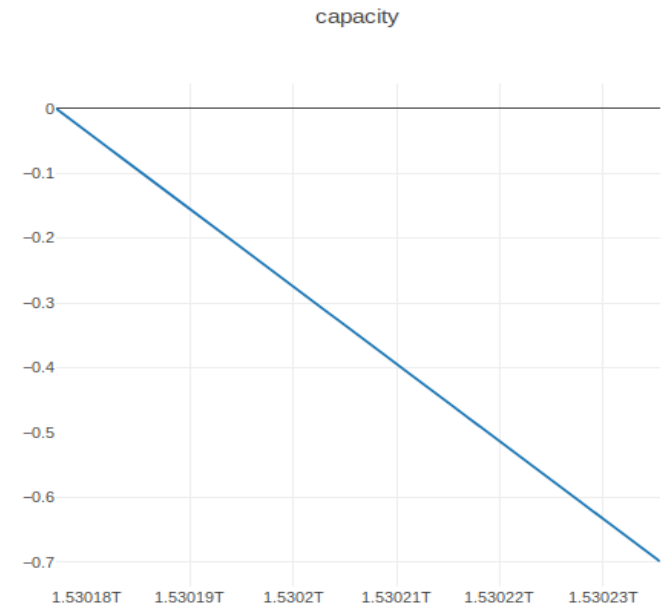

voltage_current

## ACTUAL APPROACH

- Two graphics to measure current, voltage and capacity.
- Posicionate moise over the graphic to know exactly data.
- Posibility to export to png, pdf…
- Zoom in, zoom aut

capacity




## First approach

- Make contact dash libray
- Two decimal precision

## Next implementation

- Interactive visualization
- Include checkboxes
- Include buttons

# Python libraries in a virtual environment

## Utilities

Installation

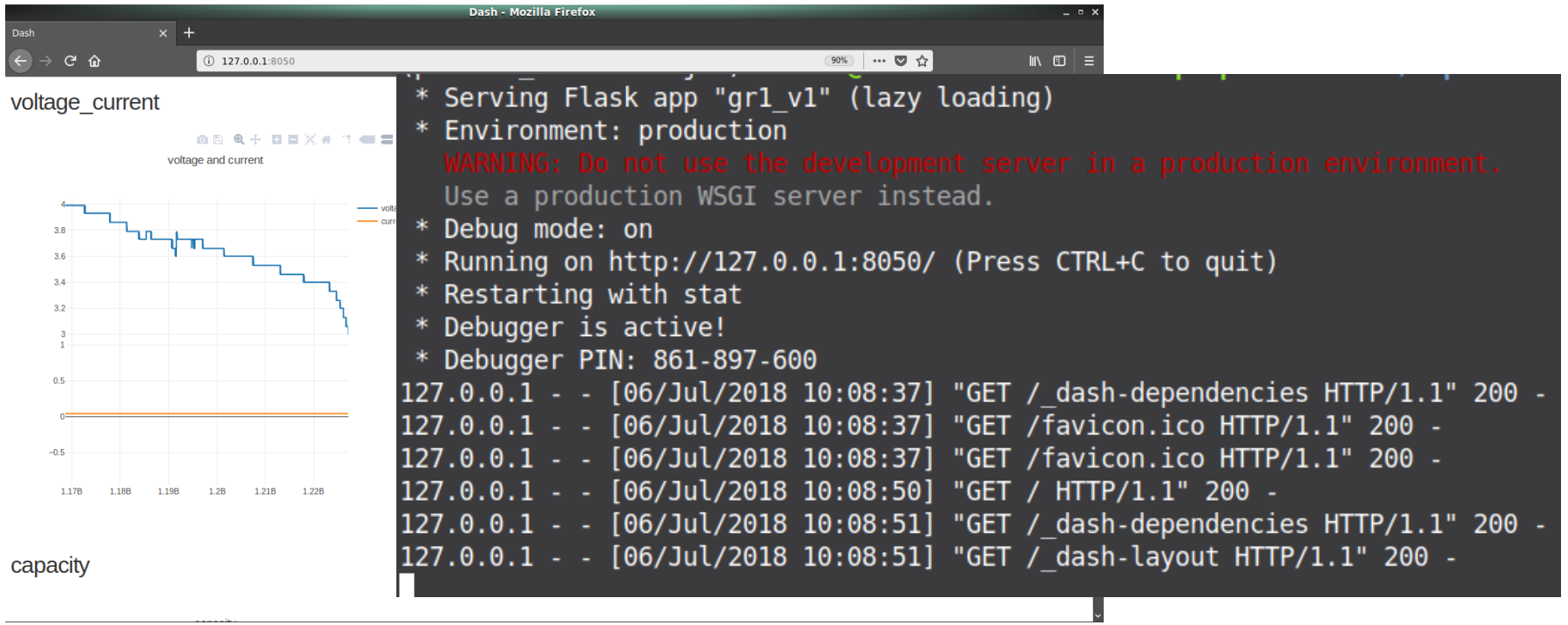
python3 -m pipenv shell

## Libraries

**Dash** create our own server

**Pandas**: work with data

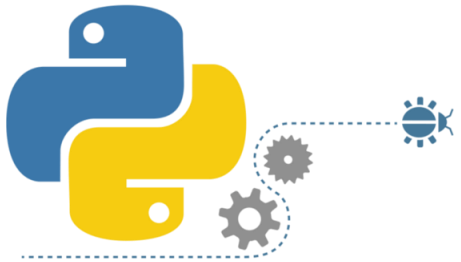
**Plotly**: graphic design




## Design proyect

- Select test type
- Charge.csv data sheet
- View result in real time

- Stablish multimeter comunication
- And control it
- Error management



# That is all
# HAVE YOU ANY QUERY?