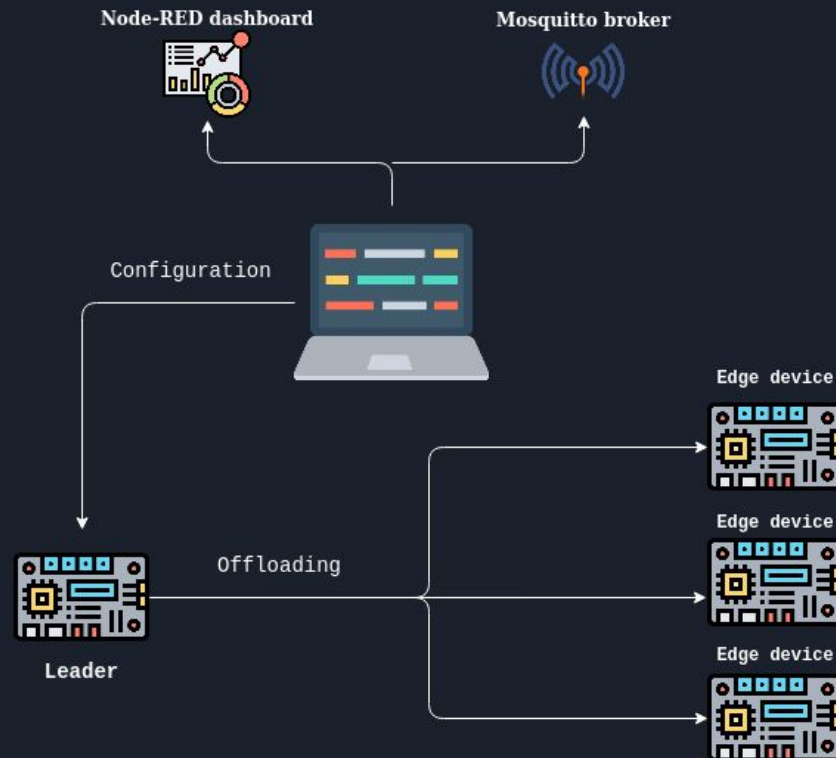
The background is a dark blue gradient. In the top-left corner, there are two overlapping geometric shapes: a blue parallelogram and a light green parallelogram. In the bottom-left corner, there is a circular inset showing a close-up of a circuit board with various electronic components. In the top-right corner, there is a faint, stylized pattern of interconnected lines and squares, resembling a circuit or data flow.

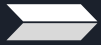
Framework to load  
balance data analysis  
on the device or at  
the edge

# System diagram





# Tools



Mosquitto (mqtt)



Node-Red (dashboard)



VSCode with PlatformIO (embedded development)



paho mqtt (python library)



# Flowchart

---

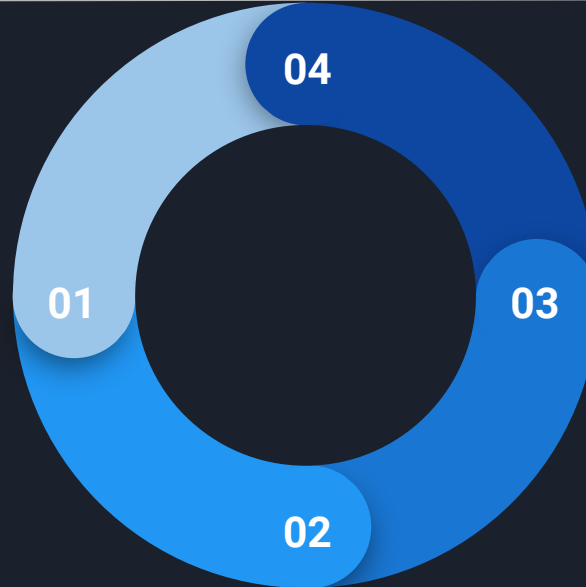
## Configure the problem

Choose the type of problem, its input and the number of iterations per second

---

## Load balancing

Determines if and how to assign load to edge devices



## Show results

View results and performance on the dashboard

---

## Run the problem

Each device performs the required problem

---



# System's features

- 01 Dynamic Wifi credentials
- 02 OTA Updates
- 03 Ability to detect the computational capabilities of any device
- 04 Distribute computation to edge devices if the computational capabilities of the leader aren't enough



# System's features

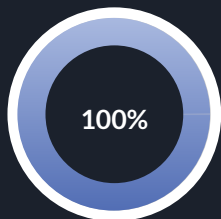
- 05 It works fine even if an edge device runs out of battery or disconnects from the network
- 06 Monitor the status of the mqtt broker constantly
- 07 Dashboard to monitor workload distribution
- 08 Stop the system in case of too high latency



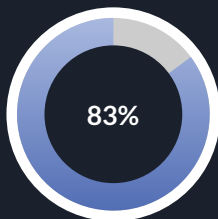
# Load distribution

As long as the leader is able to solve the number of iterations of the required problem, then no other device is involved.

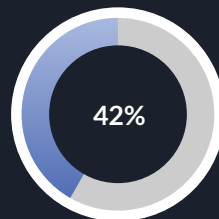
Otherwise, the leader carries out all the iterations he can and distributes them to the edge devices based on their computational capabilities: the more powerful a device is, the higher the number of iterations assigned to it will be



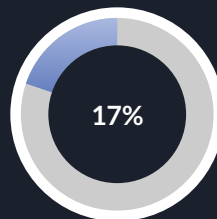
Leader



Edge1



Edge2



Edge3

# Example

≡ Task

## Vector multiplication

Vector A \*

Vector B \*

SUBMIT

CANCEL

## Prime number

Number \*

SUBMIT

CANCEL

⏏ STOP TASK

## Word Count

Text \*

SUBMIT

CANCEL

Num sec

45850

Result: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47

## Performance

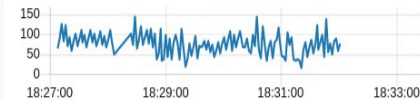
### Byte received by mosquitto broker



### Byte sent by mosquitto broker



### Latency (ms)

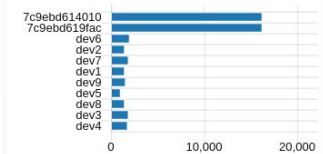


CLEAR GRAPHS

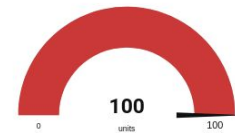
Max latency

## Load chart

### Device Load Bar (num\_sec)



### Leader load (%)



### Byte store in queues by mosquitto broker







Thanks for the attention