

Smart city

What a smart city could be like, integrating IoT sensors and scenarios to further enhance the convenience of the city's services for the inhabitants, while improving its energetic efficiency.



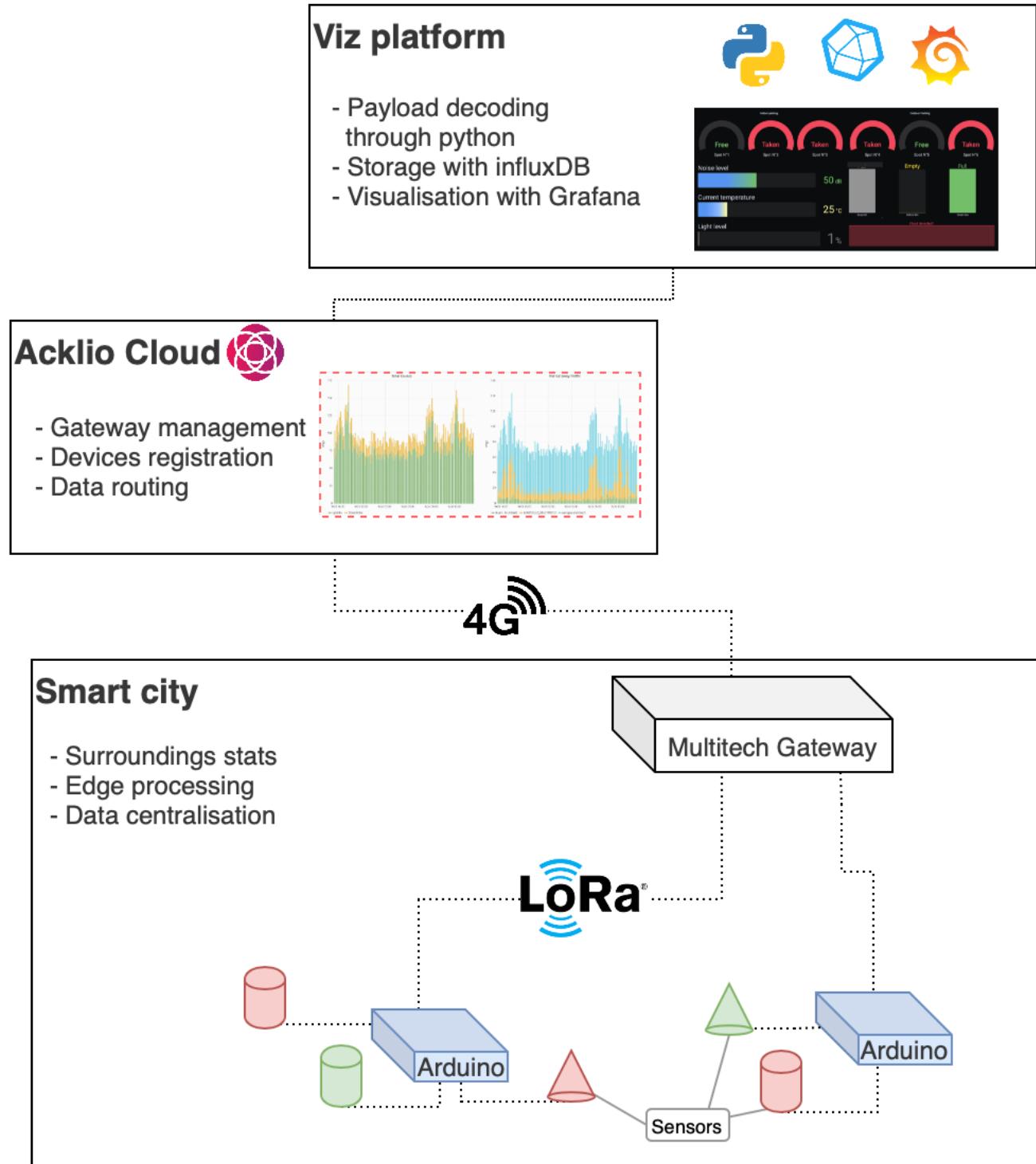
- Smart city
 - Introduction
 - Components overview
 - Hardware listing
 - Acklio Cloud
 - Cloud platform
 - More infos
 - License

Introduction

This is the repo for everything regarding the Axians's smart city model.

Components overview

Here is a quick diagram showing every major component of the model :



- Various sensors (temperature, humidity, noise, parking spots..) are connected to Arduinos equipped with LoRa antennas, that transmits to a nearby 4G LoRa gateway.
- Data & management informations are uploaded to our Acklio server.
- Actual sensor data is retrieved by an AWS instance for decoding/storage/visualisation.

Each section below is describing one of those components.

Hardware listing

Here is a list of the hardware components used in the smart city model :

- 3 [Arduino Uno](#) as micro-controllers, on top of which sits:

- A [Grove Hat](#) for easy cable management, with the following sensors :
 - [Temperature sensor](#),
 - [Noise sensor](#),
 - [Light sensor](#),
 - [Hall effect sensors](#),
 - [Standard LEDs](#),
 - [Chainable LEDs](#),
 - [Ultrasonic distance sensors](#),
- A [LoRaWAN antenna](#), to upload the gathered data to a nearby gateway,
- A [Multitech LoRa gateway](#), to receive the payloads from the antenna

Each arduino is responsible for a specific task :

- Arduino 1 is managing the street lamps and monitoring the temperature / noise level of the city,
- Arduino 2 is managing the city's trash cans
- Arduino 3 is managing the parking spots

Acklio Cloud

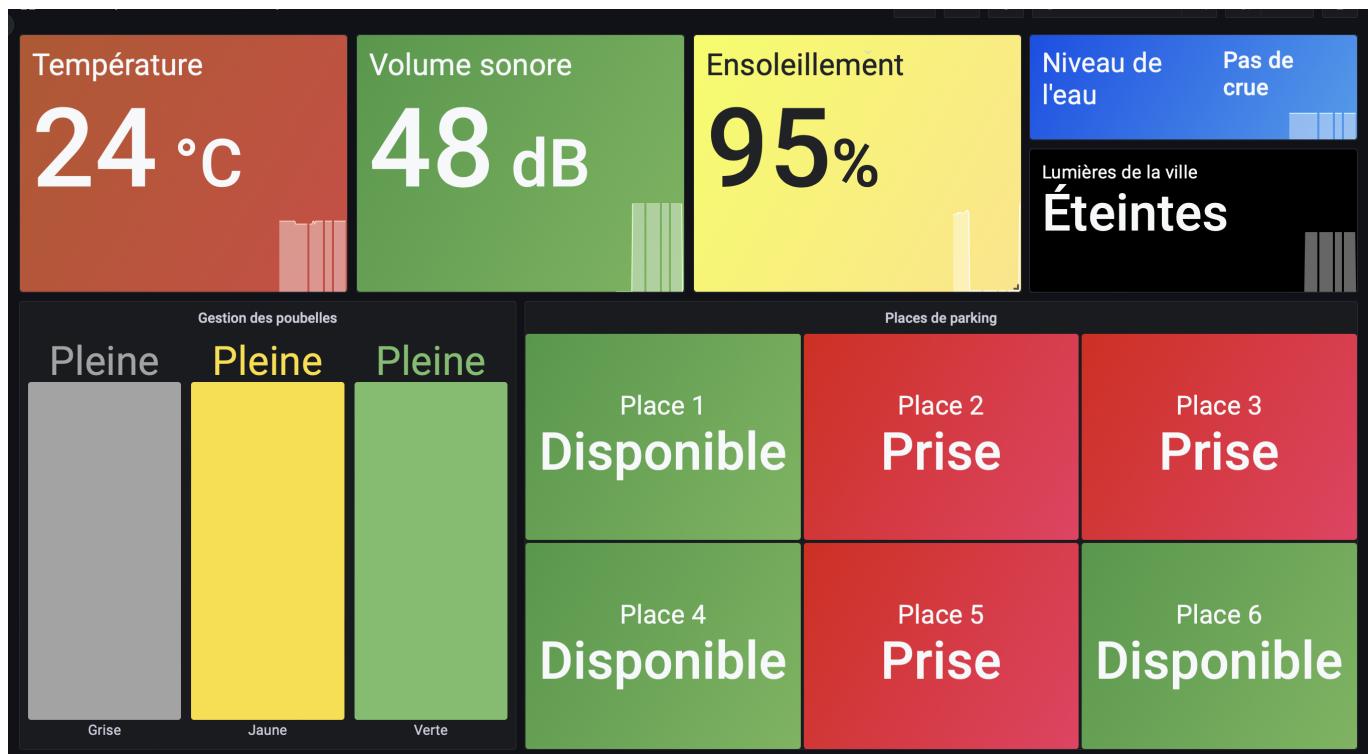
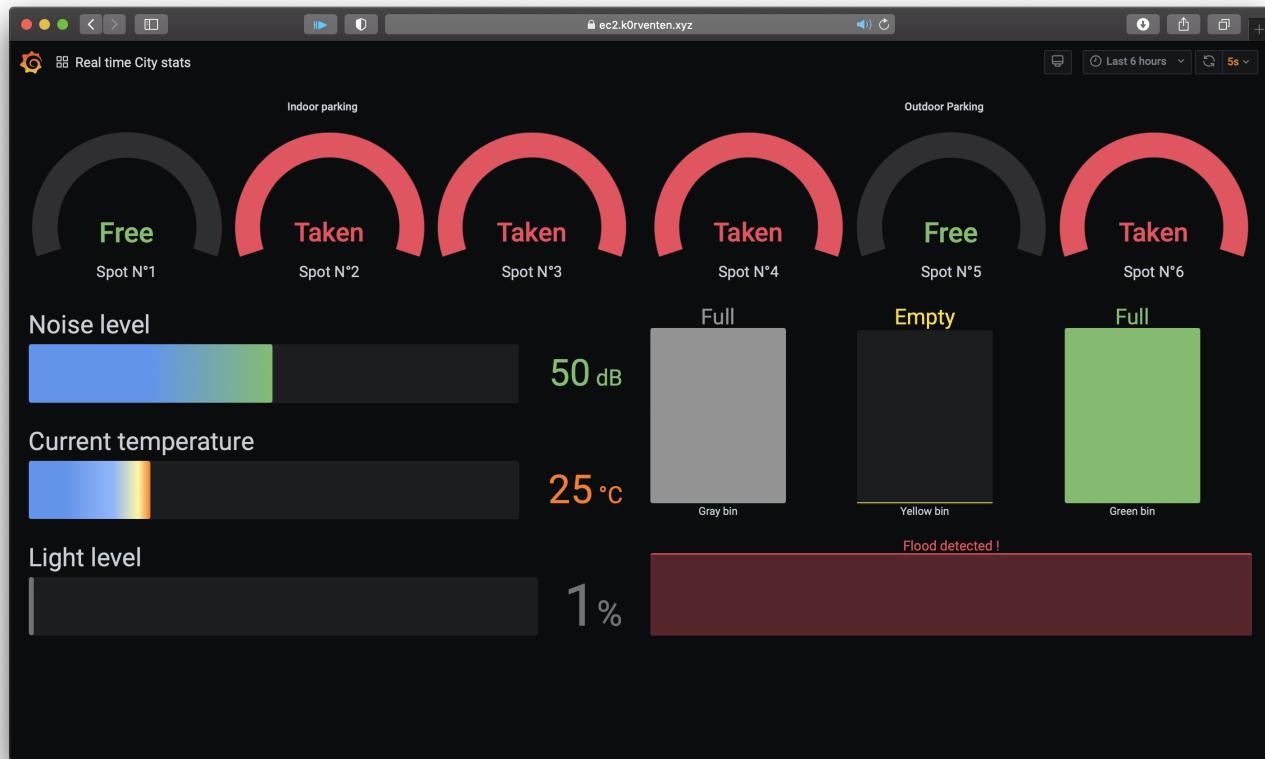
To manage our devices (gateway, sensors..) and enable 3rd party apps to access our data, we are using Acklio's cloud.

The screenshot shows the Acklio Cloud platform interface. At the top, there is a navigation bar with links: Dashboard, Devices (which is highlighted in red), Device profiles, Gateways, Connectors, Channels, Users, Acklio Observatory, Service Desk, and Logout. Below the navigation bar, it says "LNS v.2.9.0". Underneath, there is a sub-navigation bar with links: Registry (which is underlined in red), Connected, and Discovered. On the right side of this bar are two buttons: "UPLOAD DEVICES" with a cloud icon and "ADD DEVICE" in a blue button. The main area below these bars is currently empty, showing a light gray background.

Cloud platform

A cloud-based visualisation platform is being used. Running anywhere is a influxDB + Grafana combo for long-term storage & visualisation.

Example of dashboards



it is composed of :

- a python worker that connects to Acklio, decode the payload and stores it into influxDB
- influxDB, a timeseries database for storing the data uploaded by the smart city
- grafana, to visualize the data either in "real" time, or historically.

More infos

The code, infrastructure & other components docs are documented in [DEVELOP.md](#).

License

2019-2022, k0rventen

The LoRa library is GPL, as is whatever code calling it, including the `/src` dir. But my lib for the sensors in `/lib/sensors` is under MIT. Same goes for the code of the python worker, all MIT.