Design and Implementation of an Environmental Monitoring System

Felipe A. Moreno
Advisor: Prof. Manuel Castillo

UNIVERSIDAD NACIONAL DE INGENIERÍA | VICERRECTORADO DE INVESTIGACIÓN

PRONABEC

CTIC

About me

B.Sc. Felipe A. Moreno
www.fmorenovr.com

Content

- Motivation
- Related Works
- Methodology
  - Technologies
  - System Analysis

- Application
  - Web Dashboard
  - Rest API

- Conclusions
  - Main Contributions

Motivation

Motivation

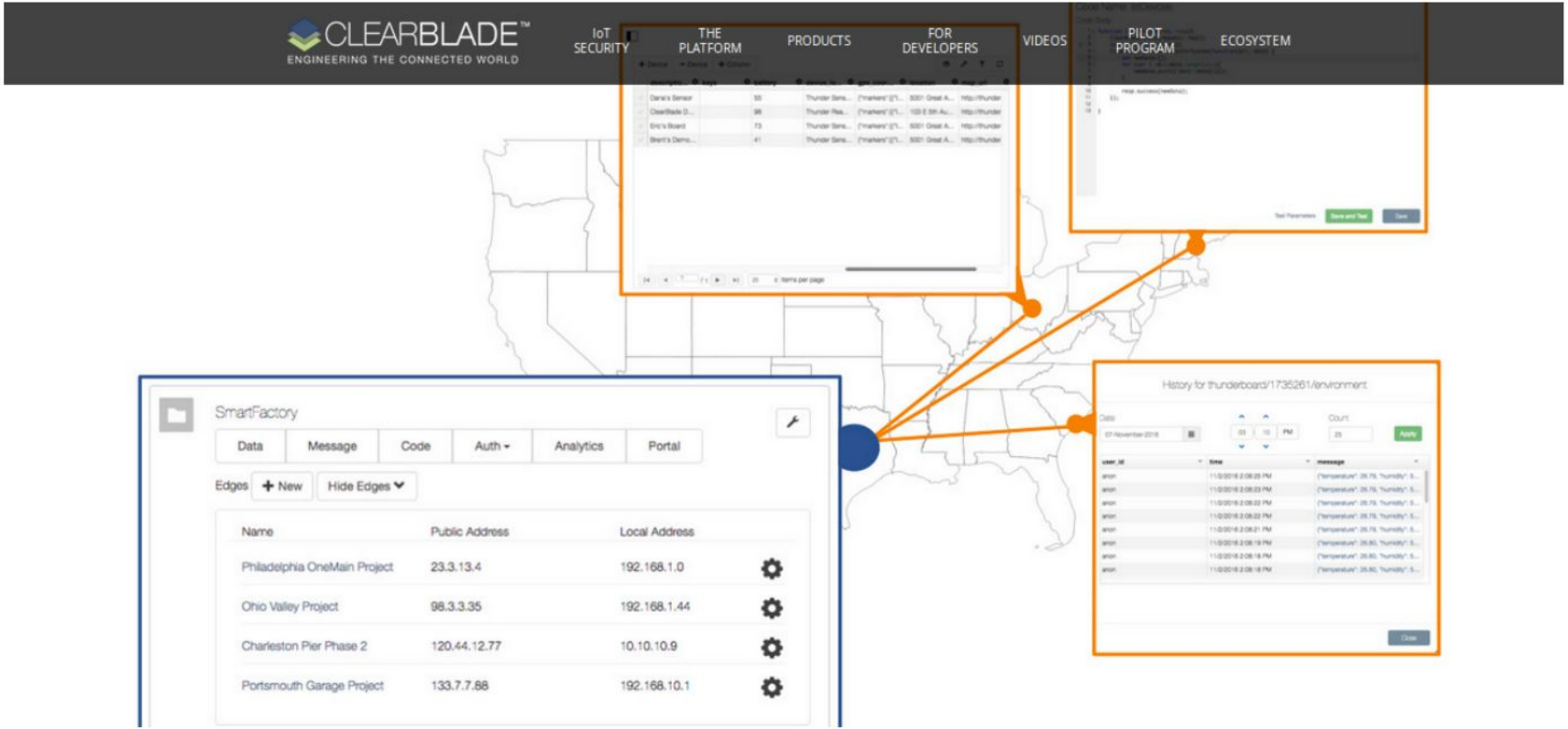
The motivation for which this theme was decided is due to the huge amount of software, libraries and frameworks that carry out this work using other technologies to obtain and update in a certain time (it could be time real), in addition to the great need for software that provides information in a friendly and easy-to-use manner.

Related Works

ClearBlade

It is a web platform software which provides services on IoT, that is, you can create connectivity with devices any, which provide any information that you you want and the ClearBlade, using the MQTT protocol and SDK of development, it will notify you about changes (on off, data, connectivity) in the associated devices, that is, ClearBlade works as a Socket MQTT web broker.

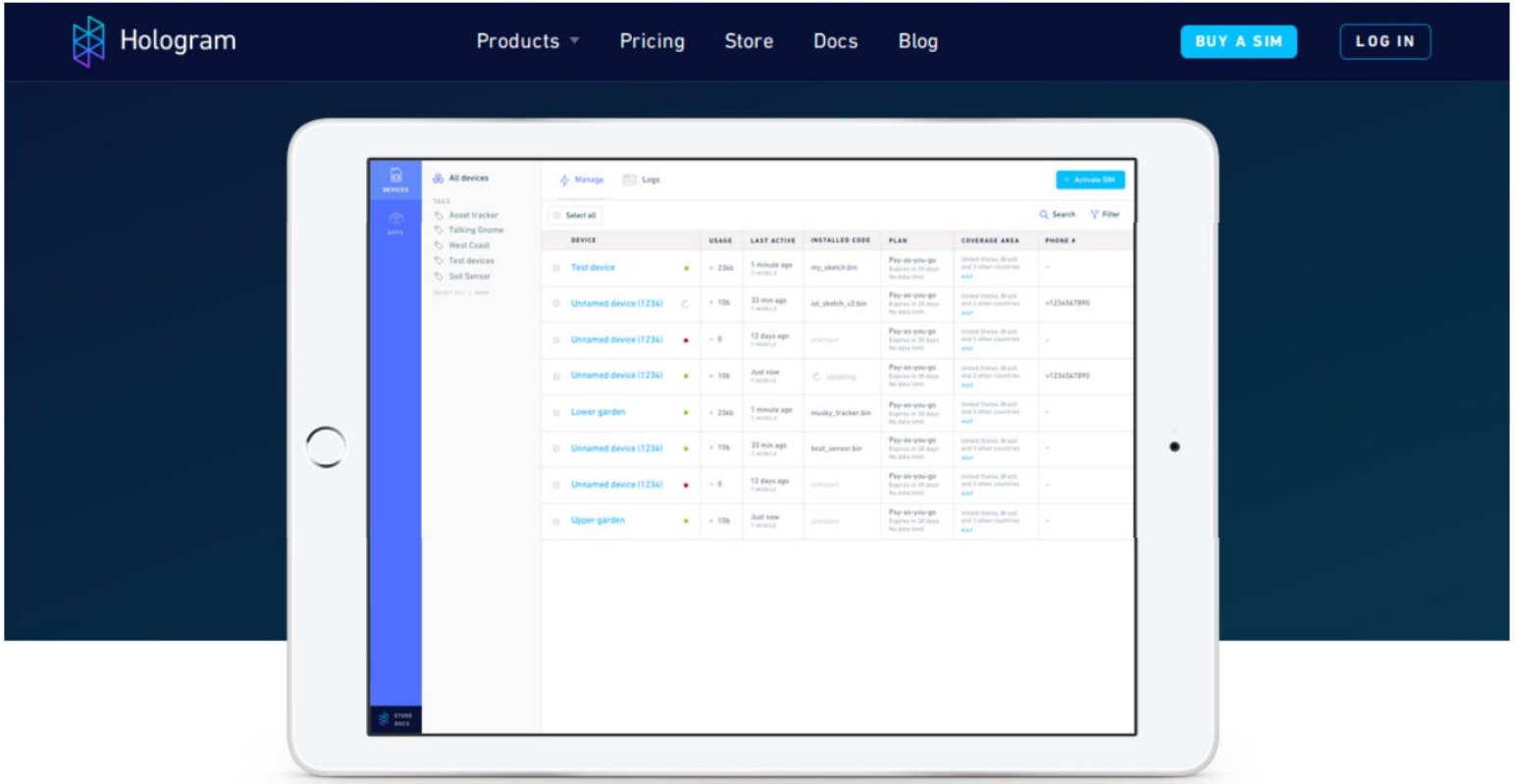
ClearBlade


CLEARBLADE™
ENGINEERING THE CONNECTED WORLD
IoT SECURITY
THE PLATFORM
PRODUCTS
FOR DEVELOPERS
VIDEOS
PILOT PROGRAM
ECOSYSTEM
SmartFactory
Data
Message
Code
Auth
Analytics
Portal
Edges
New
Hide Edges
Name
Public Address
Local Address
Philadelphia OneMain Project
23.3.13.4
192.168.1.0
Ohio Valley Project
98.3.3.35
192.168.1.44
Charleston Pier Phase 2
120.44.12.77
10.10.10.9
Portsmouth Garage Project
133.7.7.88
192.168.10.1
History for thunderboard/1735261/environment

Hologram

It is a web platform software which provides services storage for any affiliated device that has Internet connectivity for storage, the protocols that uses Hologram for shipping are bluetooth, wifi and GSM, which they send it to their web platform, storing it for subsequently display the information obtained in your dashboard.

Hologram



Methodology

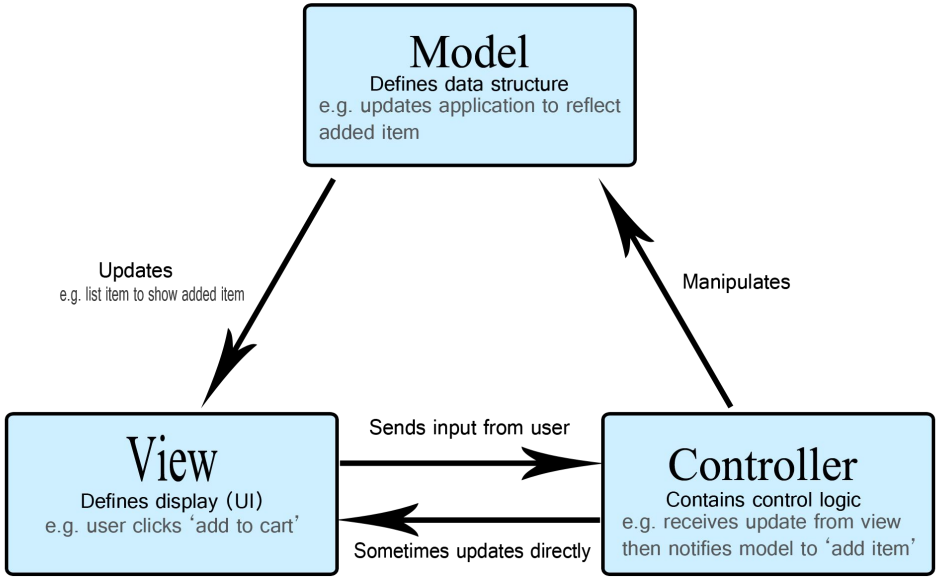
Technologies

Description

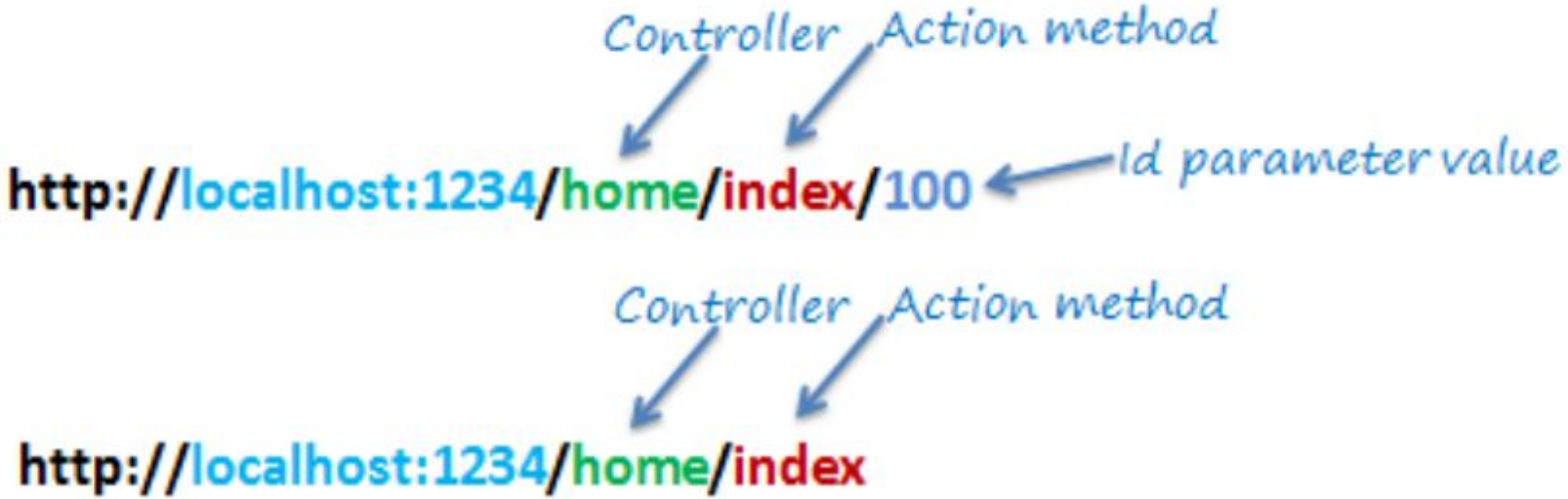
- The pattern of development on which the design of the implementation of the web platform is MVC (model-view-controller).
- For the development of a platform, a library, a package or a project in general, it is best to save the modifications or corrections of errors as they appear during the implementation stage.
- For the implementation of the database, the size of information and development of a platform, a library, a package or a project in general, it is best to save the modifications or corrections of errors according to appear during the implementation stage.
- It is an object-oriented design pattern, where objects to a class instead of the class itself instantiating the object, used for web application implementation making calls to various components necessary for the functioning.

Model View Controller (MVC)

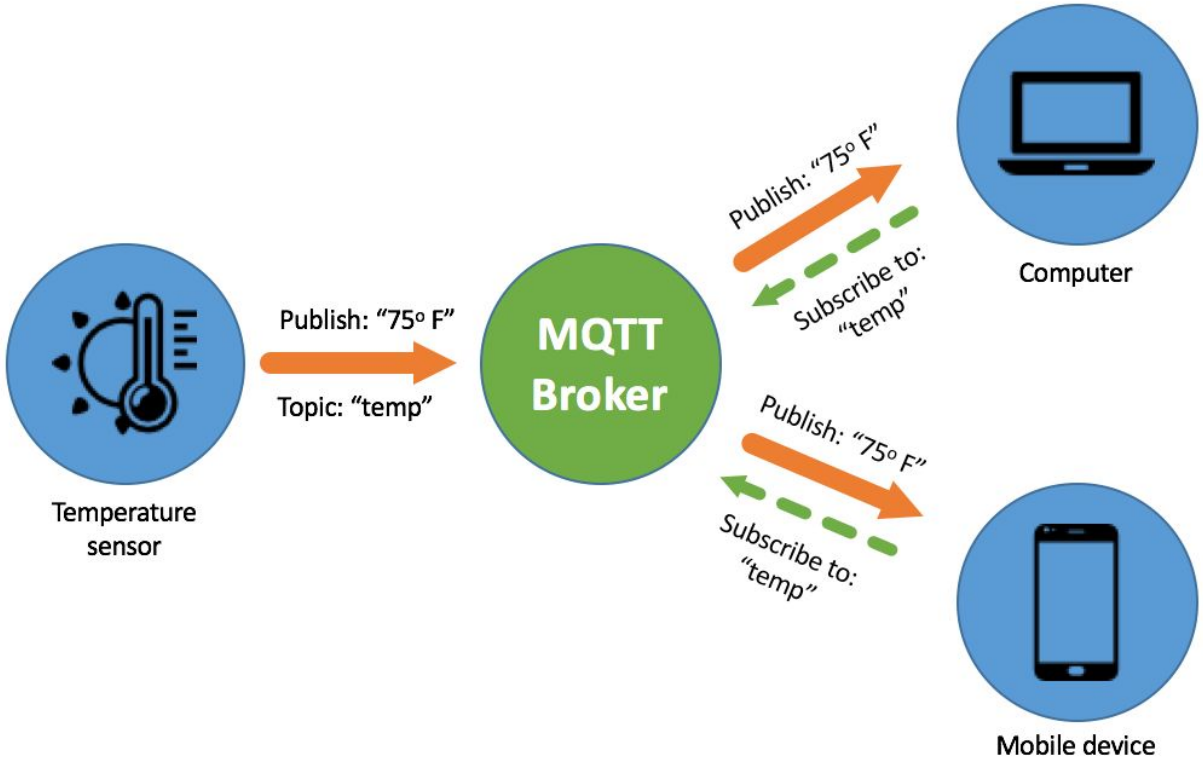
- MongoDB: NoSQL
- AngularJS
- Golang (Backend)
- github



HTTP: Routing in MVC - Rest API

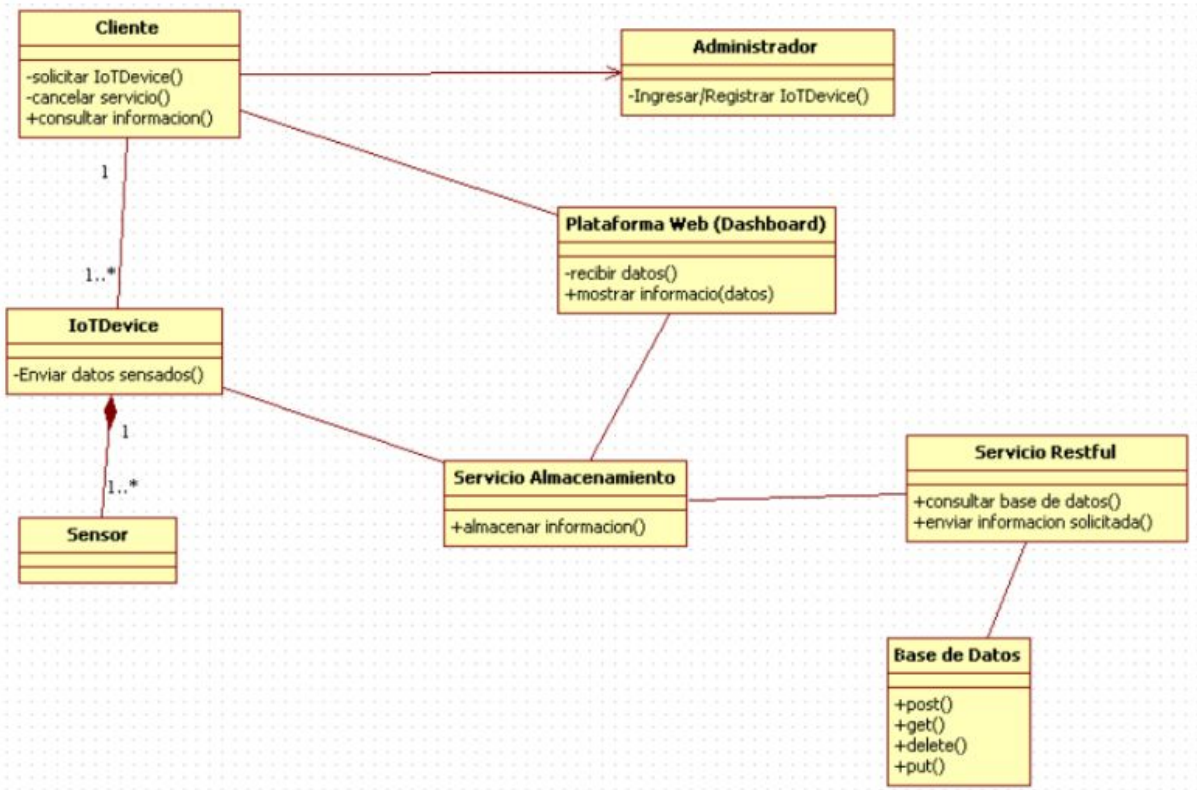


MQTT: Message Passing



System Analysis

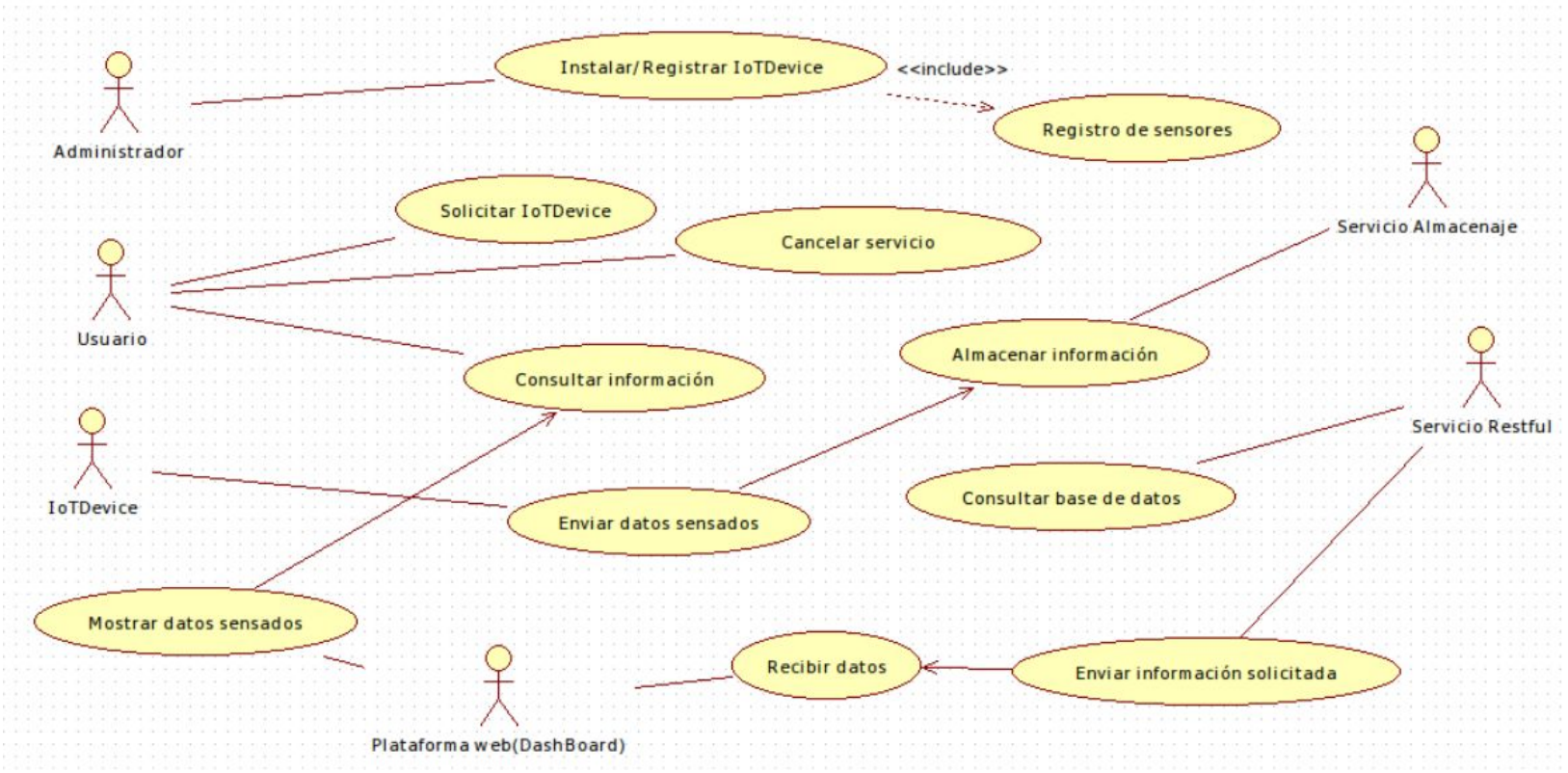
Classes Diagram



Classes Diagram

- **Administrator** It is the actor in charge of the administration of the users. He is the only user who does not have IoT-Device registered.
- **User** It is the actor that has requested a service to the system, it is that is, it is aware of the sensed information displayed by the system depending on the devices that the same user have requested.
- **IoTDevice** It is the actor that sends the information that it captures from the sensors and sends them to the storage actor.
- **Dashboard** It is the actor in charge of displaying the information sensed and stored.
- **Restful** It is the actor that is in charge of performing and returning answers according to the requests of the actor dashboard, as well as also check the storage.
- **Storage** It is the actor in charge of storing the information

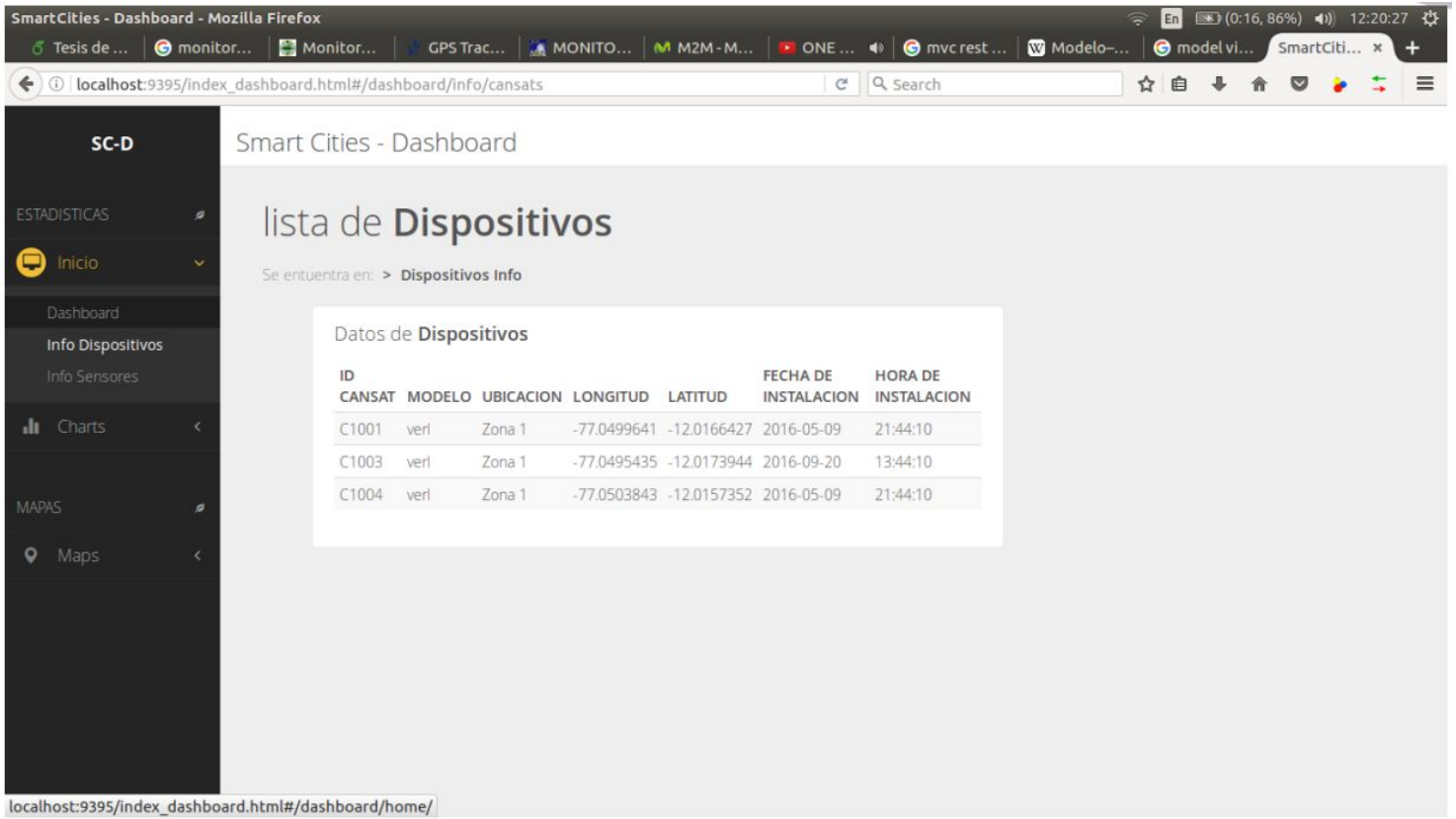
Case Study Diagram



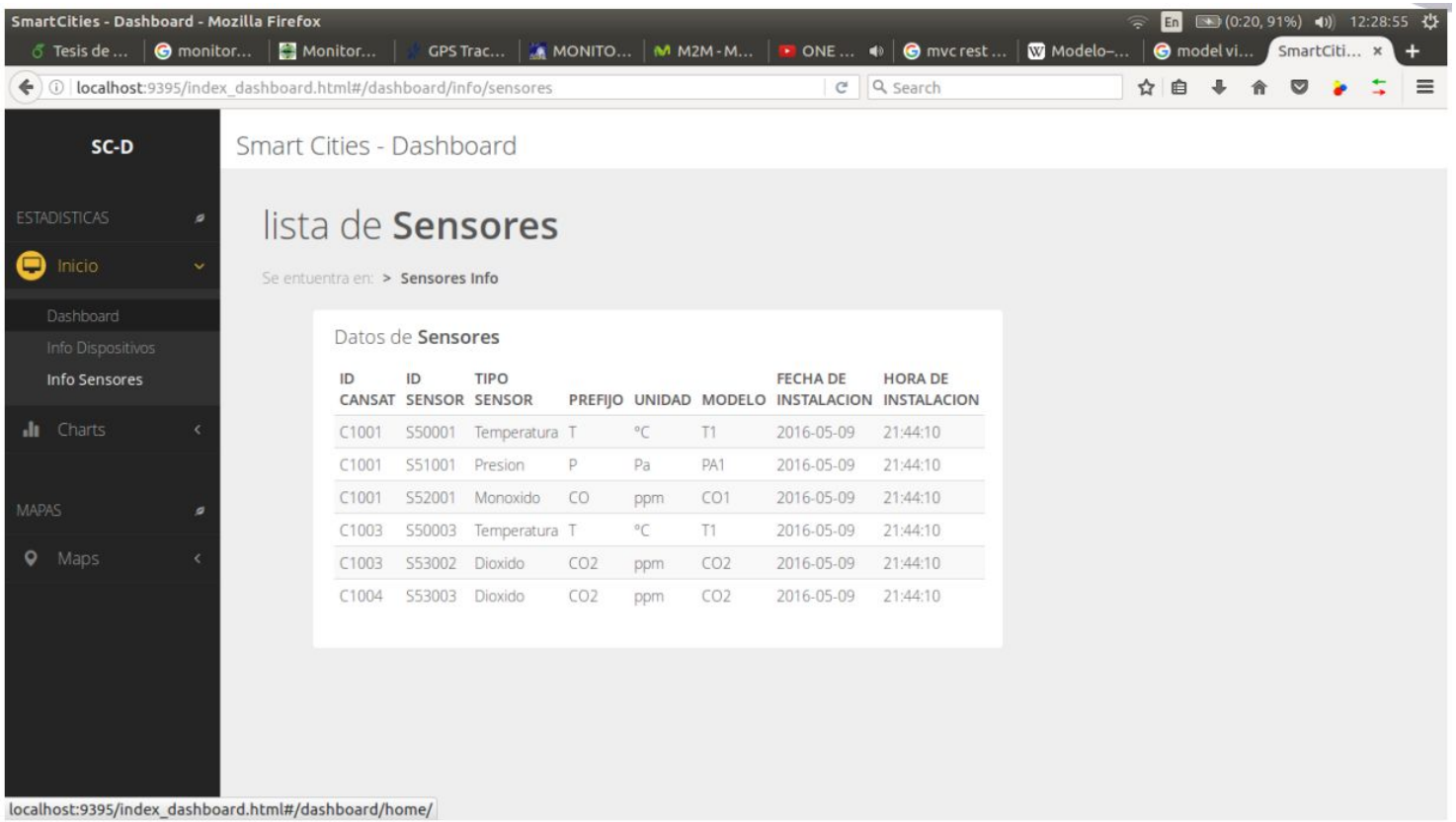
Application

Web Dashboard

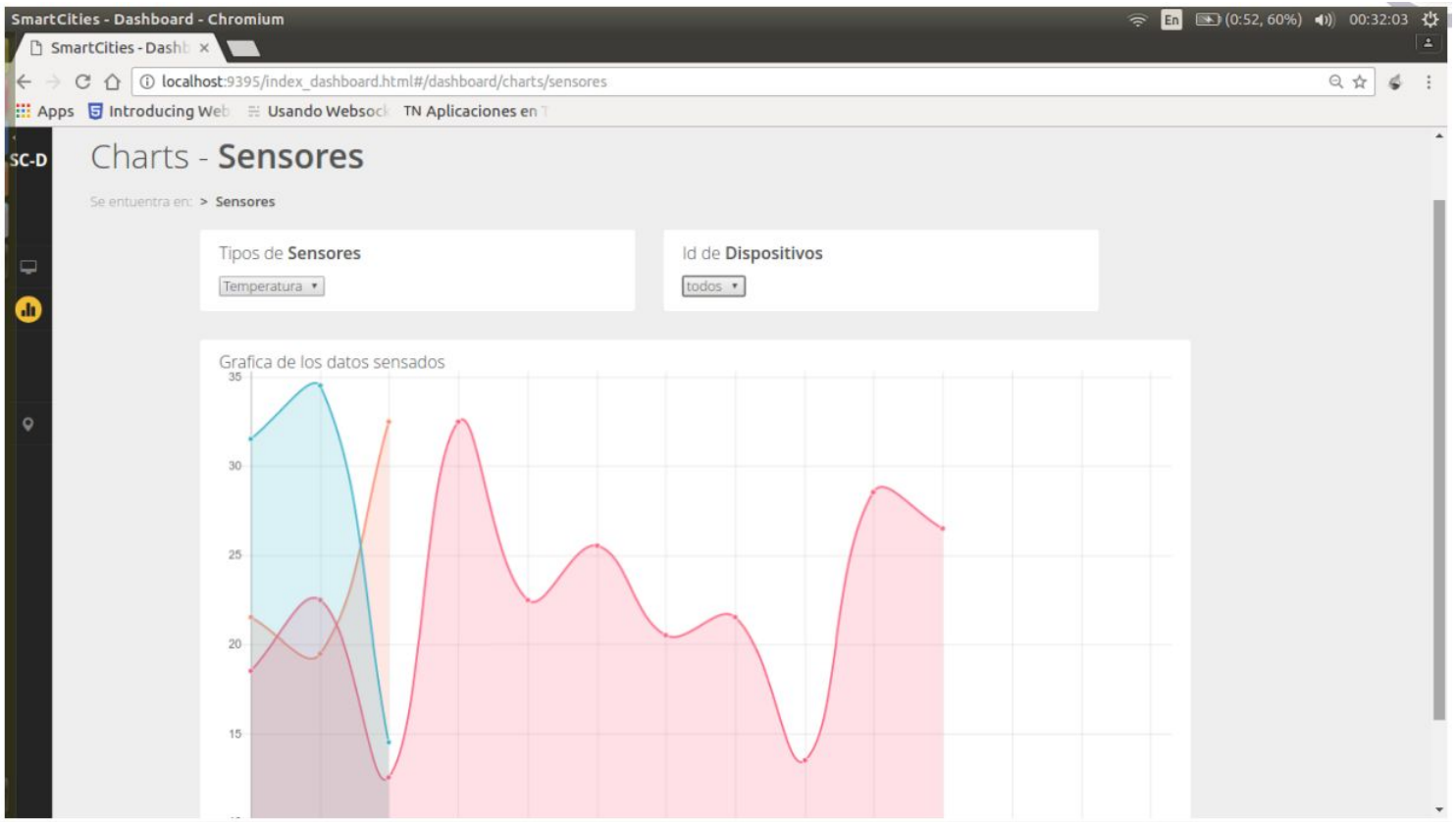
Dashboard: Device information



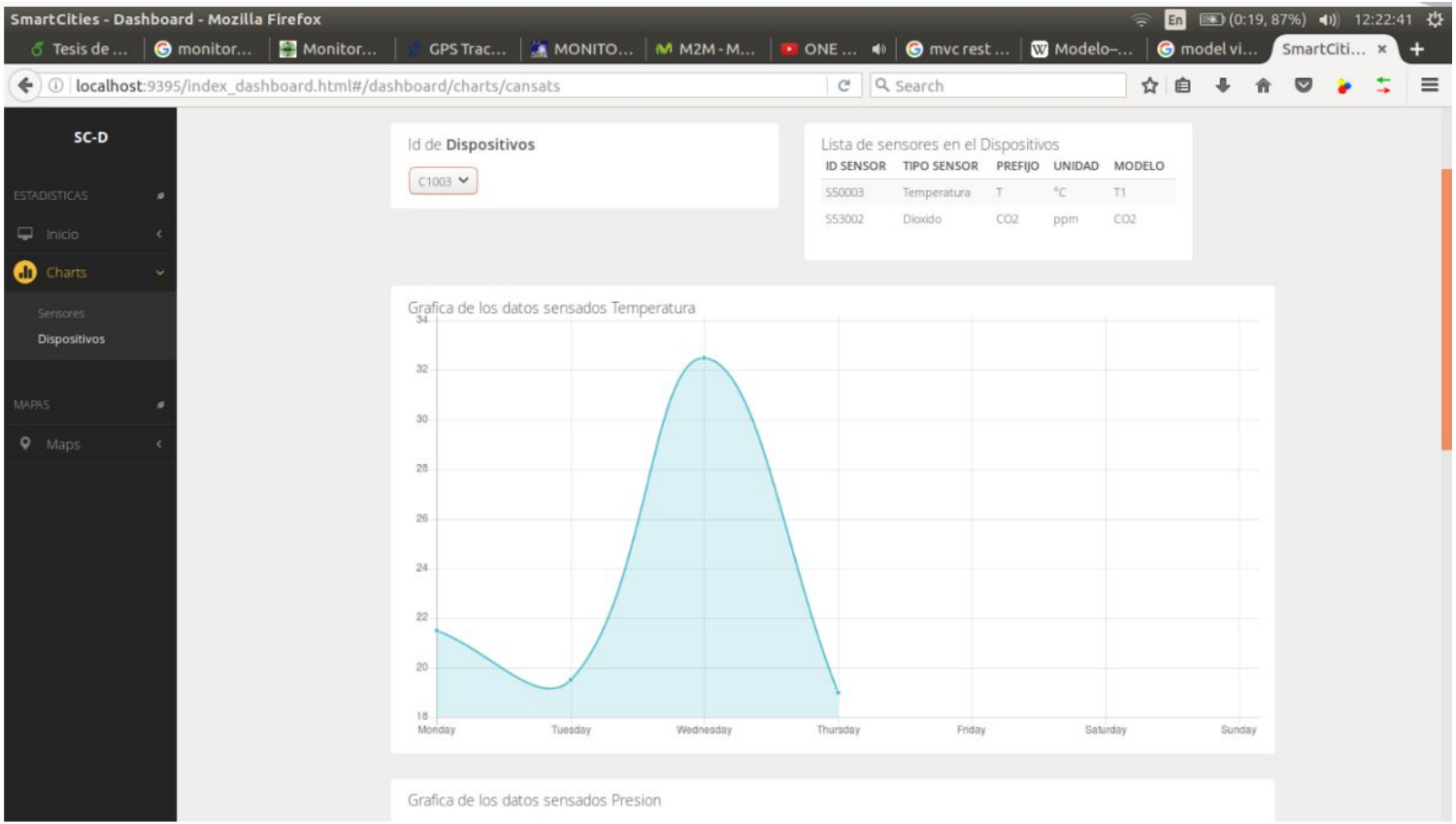
Dashboard: Device-Sensors information



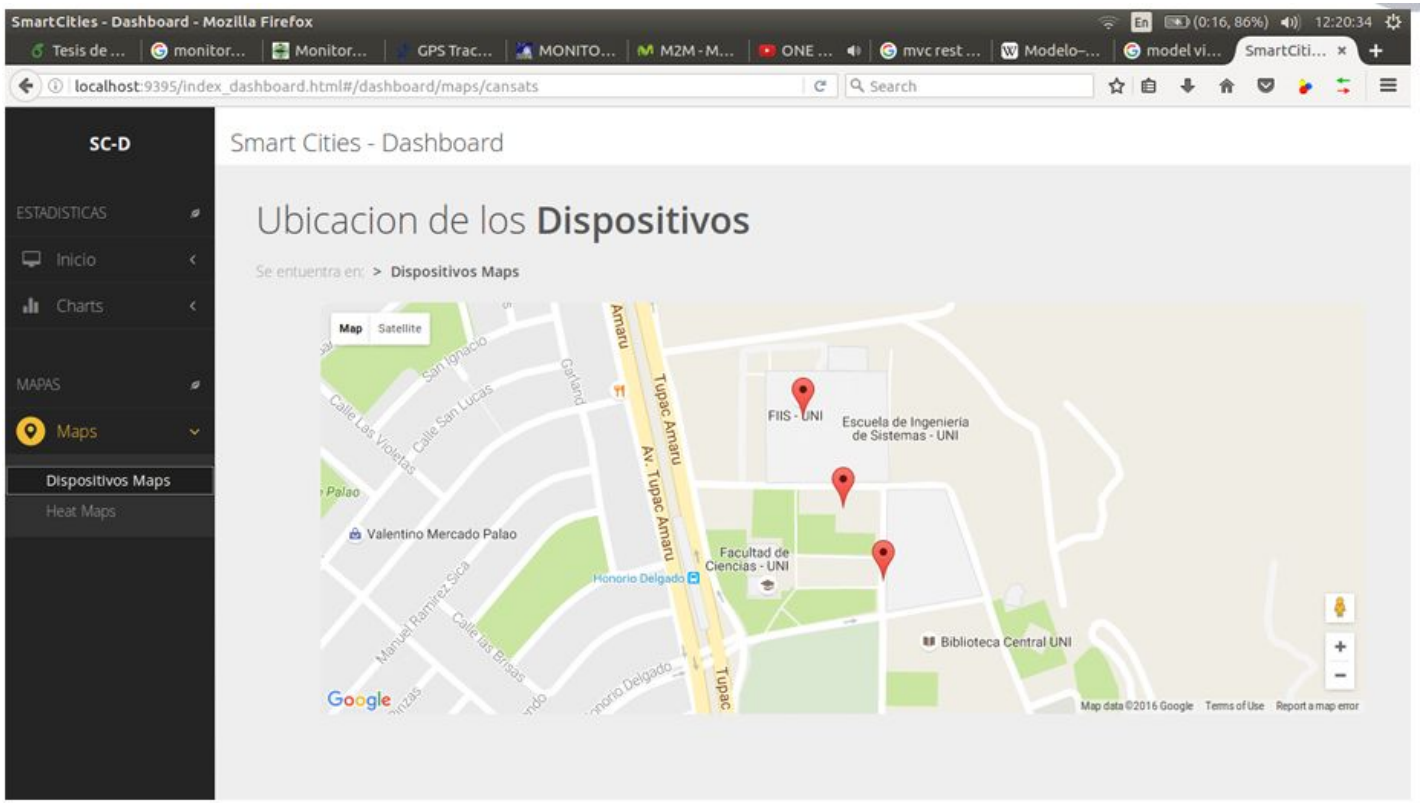
Dashboard: Device-Sensors Types information



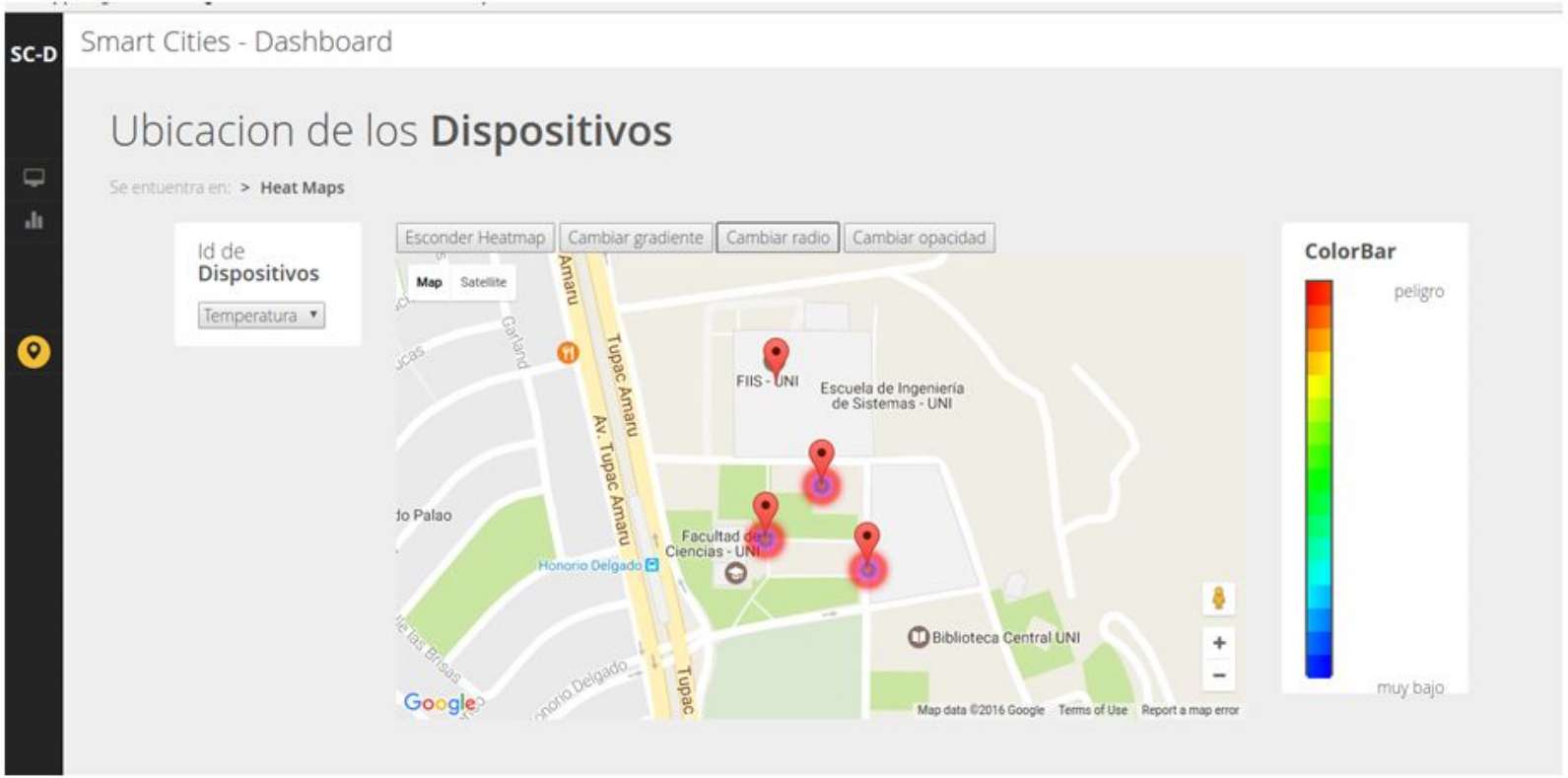
Dashboard: Sensor data



Dashboard: Sensor location

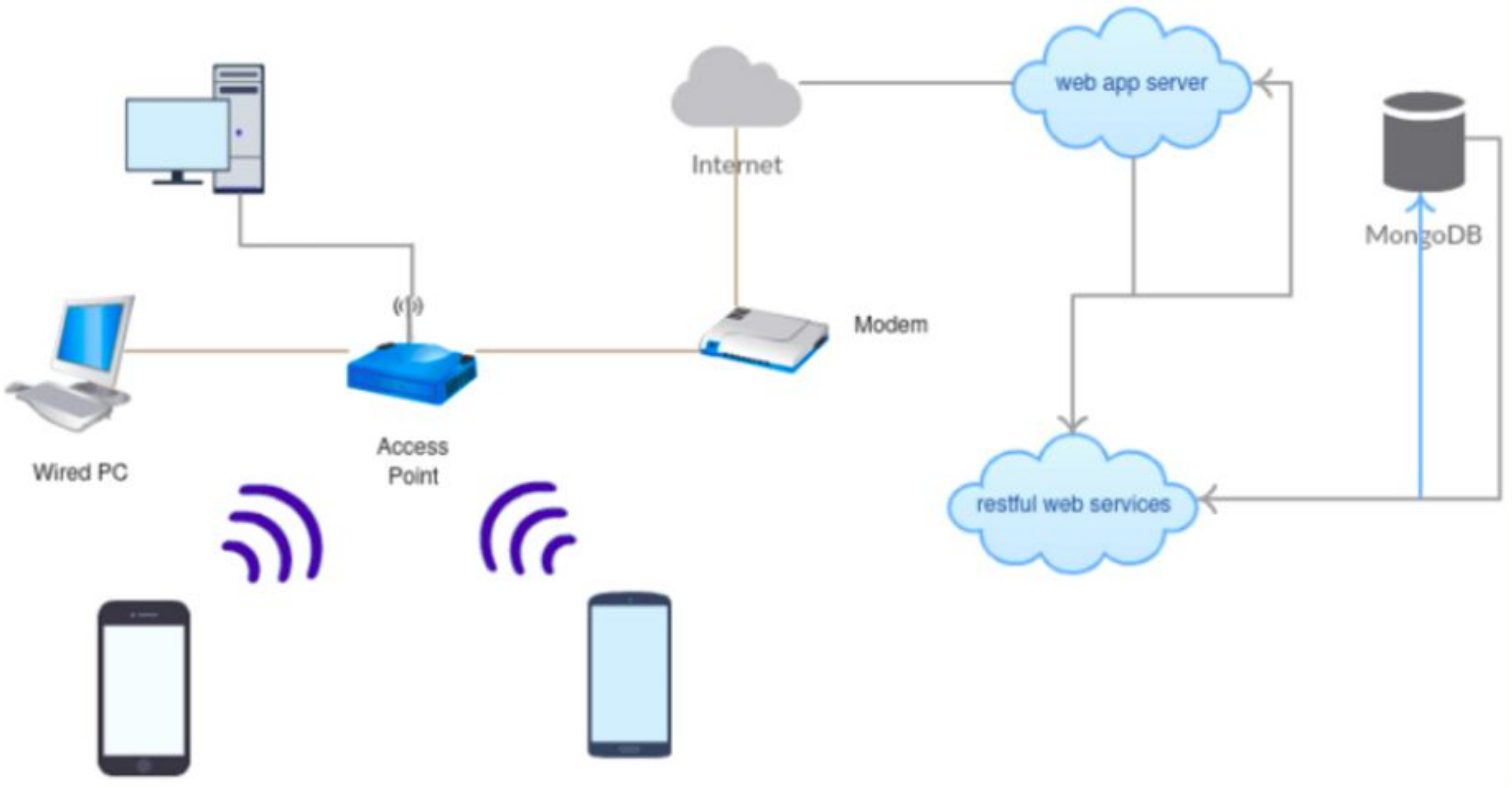


Dashboard: Sensor heatmap



Rest API

REST: System architecture



Conclusions

Main Contributions

- We propose a methodology to analyze income data from remote sensors.
- We implement a Fullstack application to process, clean, and show information sensed
- We develop a dashboard to show information in real-time.
- We build an Fog-based architecture to sensing, process, and track sensors data from different locations in real-time.

Questions?