Transport Monitoring
IoT Project

Giovanetti, La Corte, Scarrà

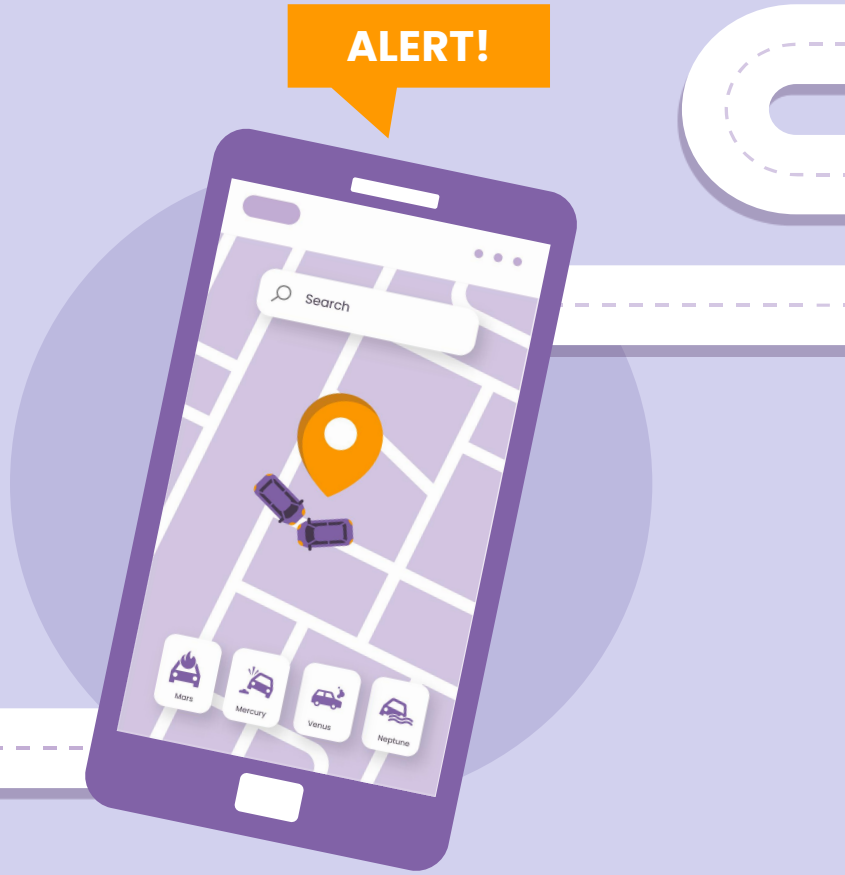


Table of contents

01

Abstract

Abstract

The idea is to build an infrastructure to monitor the quality of road trips with different vehicles, mainly buses and trains.

The aim is to track:

- Temperature
- Crowding
- Noise Pollution
- Vehicle Movements
- Punctuality

Parameters

Accelerometers

They measure changes in the bus's acceleration, helping us understand how smoothly the bus moves and if it experiences any sudden jolts or vibrations.

Noise

They assess the level of noise pollution within the bus, which can significantly impact passenger comfort.

Temperature

They ensure that the interior temperature remains comfortable for passengers throughout the journey.

Crowding

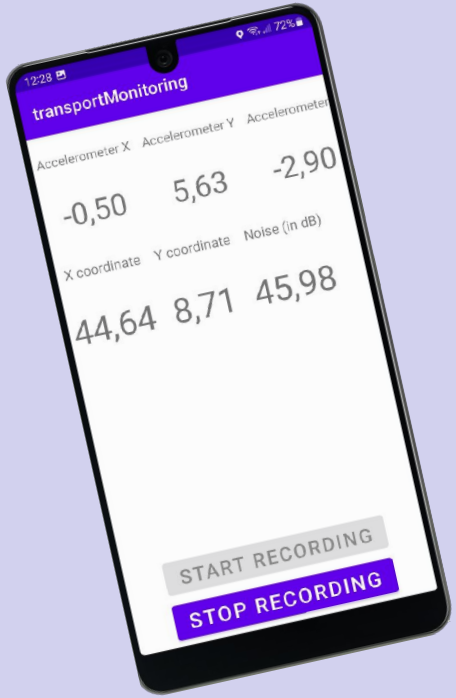
These sensors assess passenger occupancy and seating availability by providing real-time data on how crowded the bus is.

Position

They provide data on the bus's location, speed, and adherence to schedules.

02

Android



Our App

We've developed our own Android app in Java to gather data from the phone's various sensors.
Temperature and crowd data, instead, are simulated since phones don't have sensors to gather such information.
It uses the Eclipse **Paho** Java library, which manages connection and message transmission using MQTT protocol

03

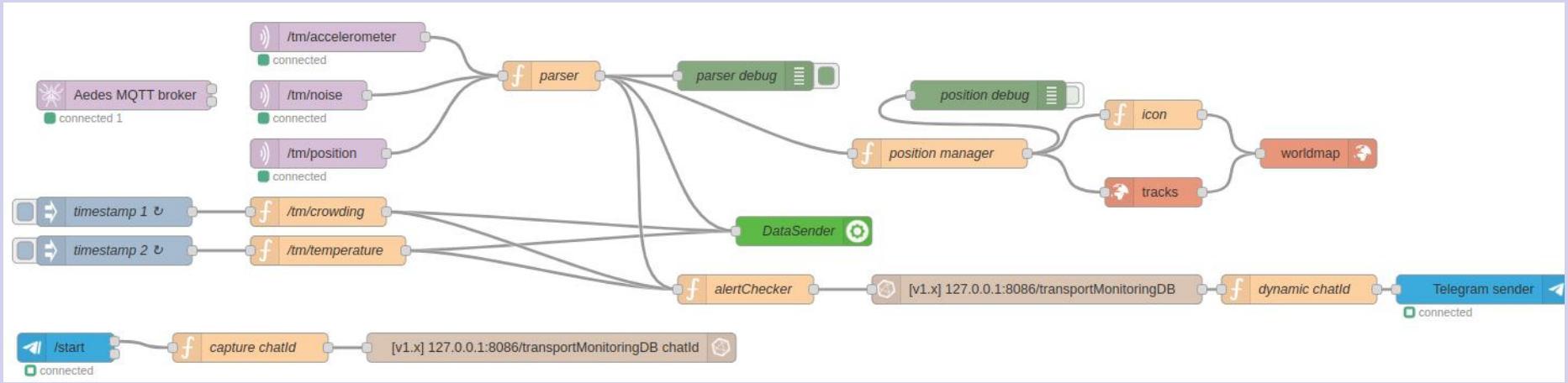
NodeRED

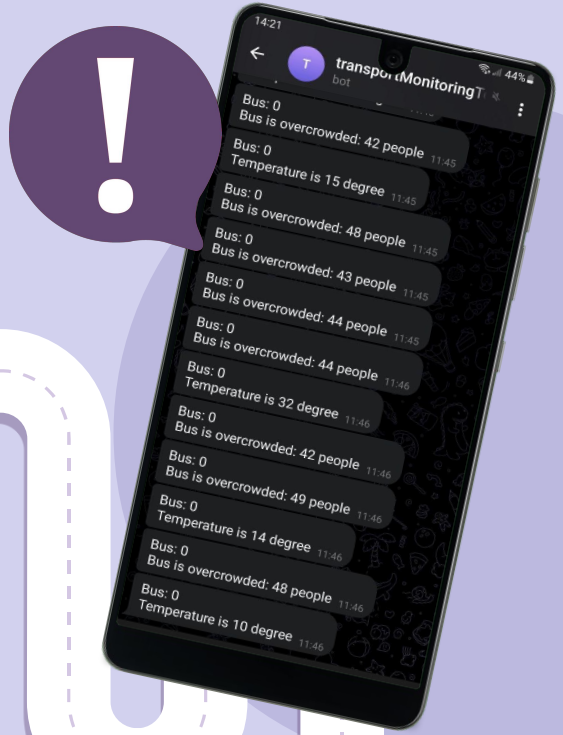
Node RED workflow

Our workflow aims to achieve different objectives:

- Simulate the crowding and temperature sensors.
- Act as an intermediary between the application, utilizing an MQTT broker node powered by aedes.js and a ThingWorx node responsible for communicating with ThingWorx via the HTTP REST protocol.
- Manage the Telegram bot to send alerts when needed.
- Display real-time location on a map

Node RED workflow





04

Telegram

We also implemented a Telegram bot through Node-RED that sends alerts if certain values fall outside a specified range.

05

ThingWorx

Entities

Thing

Representing the bus

Value Stream

Associated with the thing.

Mashups

Deriving from data of the value stream.

Alerts

Triggered when Thing's properties satisfy a specific condition.

Application Key

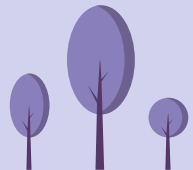
Needed to pass data from NodeRED to Thingworx.

Properties

Name	Actions	Source	Default Value	Value	Alerts	Category	Additional Info
accelerometerX				-0.06	1		m/s^2
accelerometerY				-0.04	1		m/s^2
accelerometerZ				-0.09	1		m/s^2
crowding				4	1		people
latitude				44.64	0		°
longitude				8.71	0		°
noise				24.61	1		dB
temperature				12	2		°C

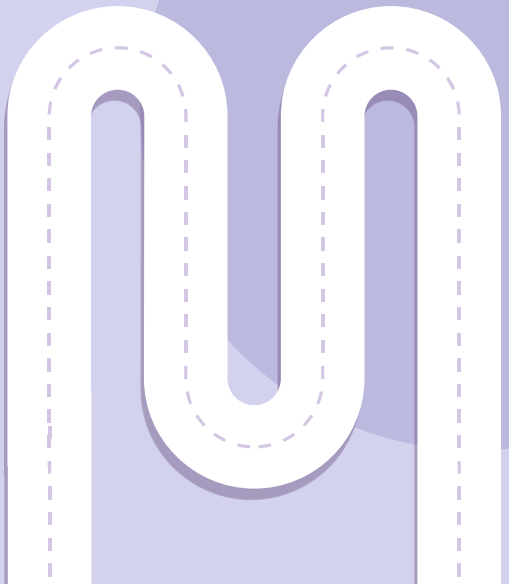
Alerts

	Name	Property	Configuration	Priority	Status	Enabled
☐	accelerometerXWarning	# accelerometerX	value <= -25 or value >= 25	1		✔
☐	accelerometerYWarning	# accelerometerY	value <= -25 or value >= 25	1		✔
☐	accelerometerZWarning	# accelerometerZ	value <= -25 or value >= 25	1		✔
☐	crowdingWarning	# crowding	value is >= 40	1		✔
☐	highTemperatureWarning	# temperature	value is >= 30	1		✔
☐	lowTemperatureWarning	# temperature	value is <= 15	1		✔
☐	noiseWarning	# noise	value is >= 70	1		✔



06

Dashboards



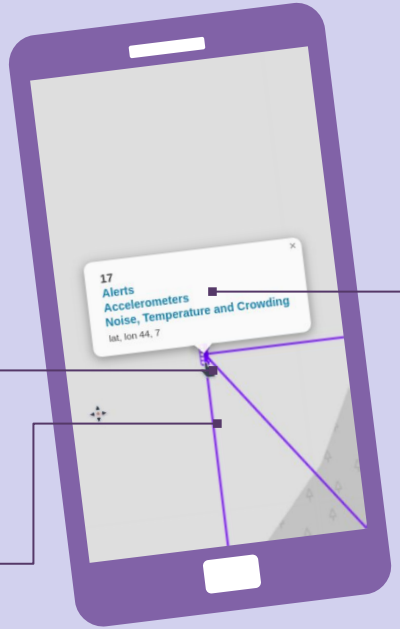
Map - NodeRED

Pointer

Icon of the vehicle moving

Movement

Line representing vehicle movements



Link to Dashboards

Links to ThingWorx Mashups

Mashups – ThingWorx

Alerts

Showing the last alert triggered for each parameter and listing the information about it.

Accelerometers

Showing Gauges and Charts of Accelerometers X, Y and Z, to monitor respectively the current value and its evolution over time.

Noise, Crowding, Temperature

Same structure of accelerometers mashup but on these three metrics.

THANKS!

Internet of Things Project 2022-23

Davide Giovanetti

Lorenzo La Corte

Davide Scarrà

UNIVERSITÀ DEGLI STUDI
DI GENOVA