VIGIWHEELS

Your autonomous sentinel

Table of contents

What makes **Vigiwheels** so **Amazing?**

A connected autonomous vehicle that **ensures the security** of your industrial building

16 000

Fire incidents on average
in industrial areas
a year

71 300

Attempts of break-in
in industrial areas
In 2018

Tahani team

Moad

Johann

Raphael

Oysho

Axel

Aïssatou

Eduardo

01

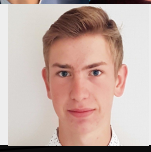
Sprint 3 summary

- Objectives
- What we have done
- Planning management

Objectives

Sprint 3 summary

01	User Interface	• Create a web page • Communication User - Vehicle
02	Fire Detection	• The car must be able to detect smoke
03	Instrument reading	• Integration of AI model in ROS Start intruder detection with QR code • Following manometers with camera



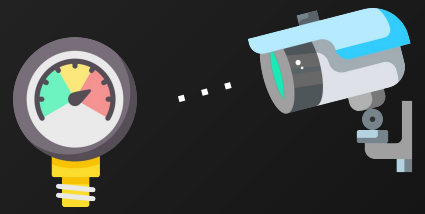
Sprint 3 summary

What we have done

- ✔ Start de user interface

- ✔ Integrate the AI to the car

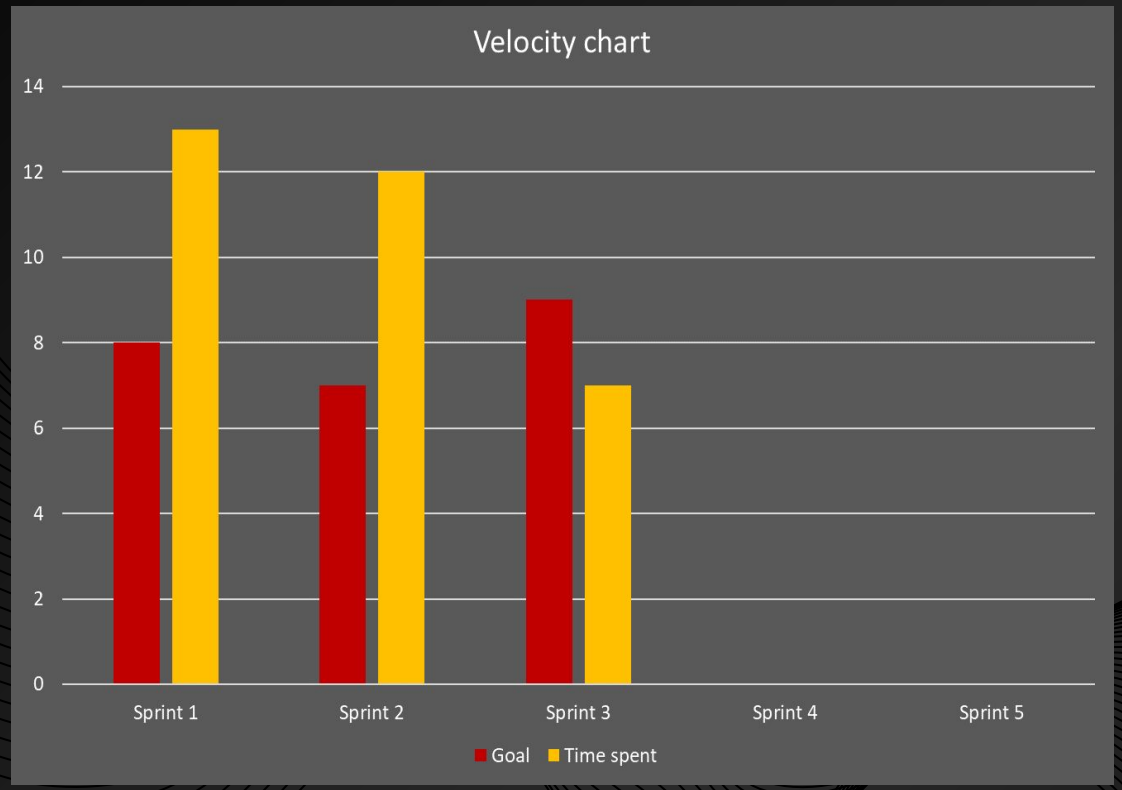
- ✔ Tracking manometers with the camera

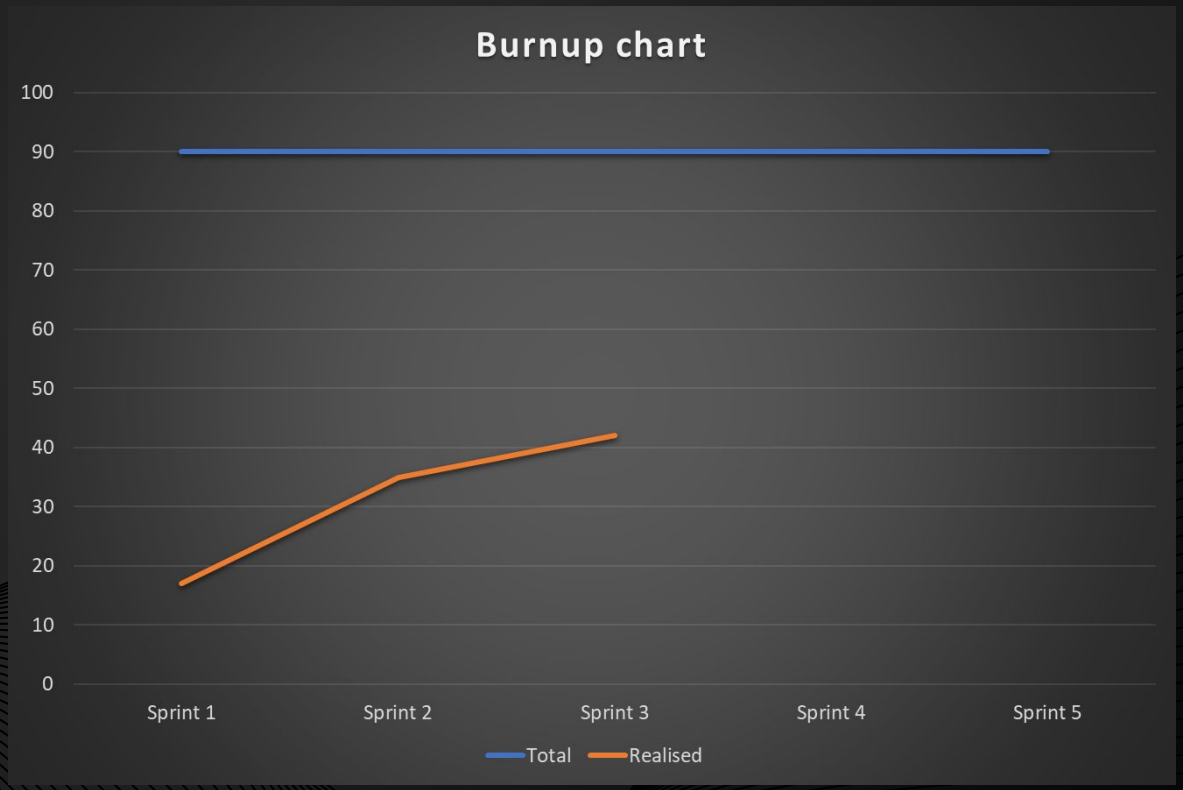


- ✔ Integrate the smoke sensor to the car

Planning management

Sprint 3 summary



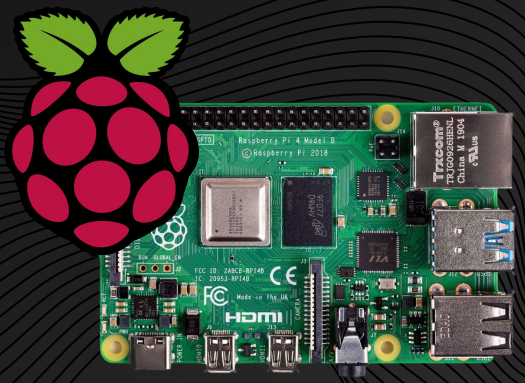


02 Demonstrations

Tracking manometers

Explanation

Ideal connection between the two cards



- ROS communication.
- Publishing images inference result.
- Subscription to obtain images from the camera.
- Predictions made by the CPU on Jetson.

Tracking manometers

Explanation

Real connection to use GPU inference

ROS

- Implementation of a local server.
- Communicate Docker as a client with Jetson's local workspace as a server.
- Resolution of hardware compatibility issues.
- Predictions made by the **GPU** on Jetson.
- The disadvantage is the use of asynchronous functions within synchronous environments.

NVIDIA JETSON

Tracking manometers

Explanation

I - AI integration in ROS

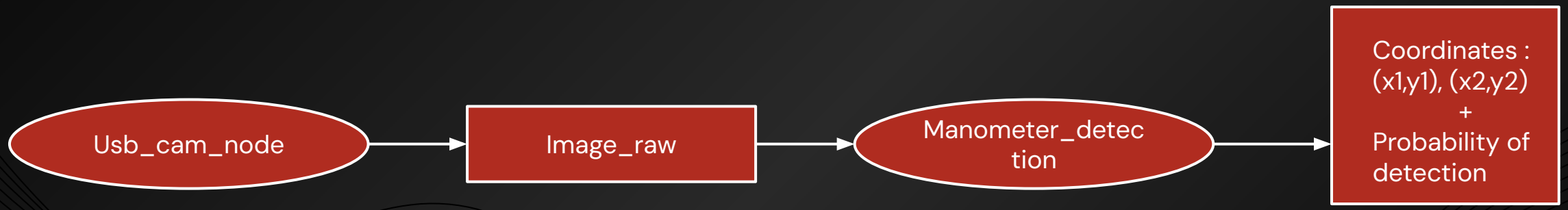


Image retrieval

Image storage

Detection of manometers in real time

Tracking manometers

Explanation

I - AI integration in ROS

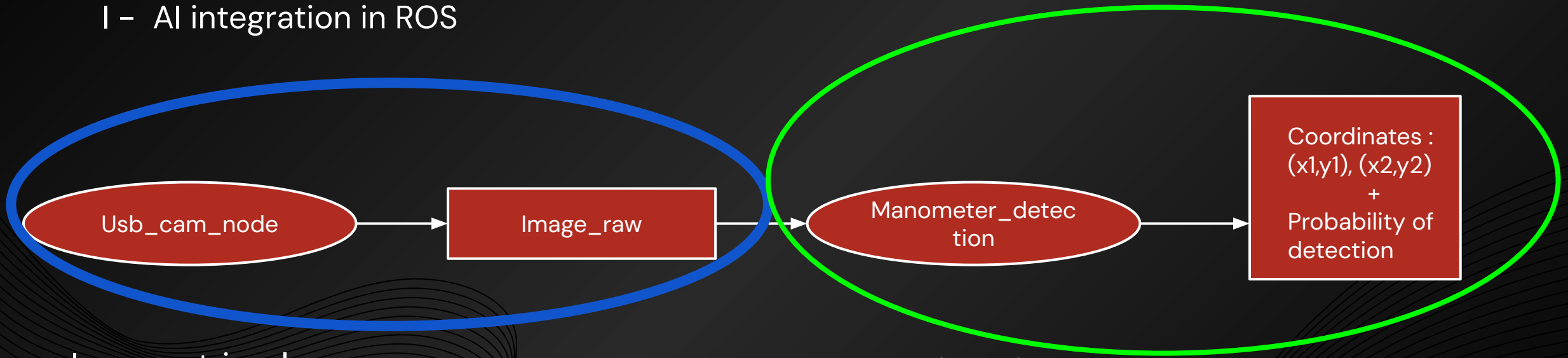


Image retrieval

Image storage

Detection of manometers in real time

Tracking manometers

Explanation

II - Follow manometer

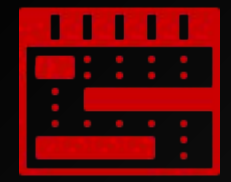
Update camera angle depending of x coordinates

Tracking manometers

Demonstration

- When the tracking mode is active, the camera follow a manometer while the car move at maximum speed
- If a manometer enter in the field of view of the camera, the camera center it

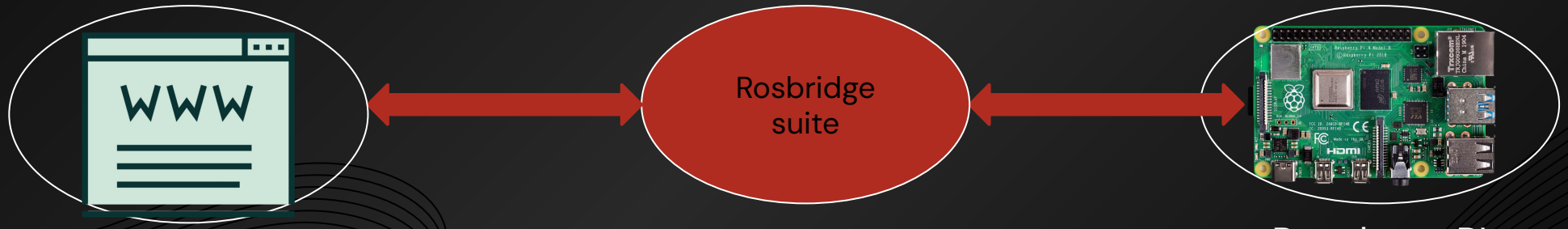
Not yet fully functional, because of the inference speed



Car Dashboard

Explanation

➢ How the connection works



RaspberryPI

➔ Rosbridge_suite is a collection of packages that implement the rosbridge protocol and provides a WebSocket transport layer

➔ The Web Interface is built with html, javascript, and css files that use images for design and user comprehension

Done & To-Do

Car Dashboard

- What is already done?

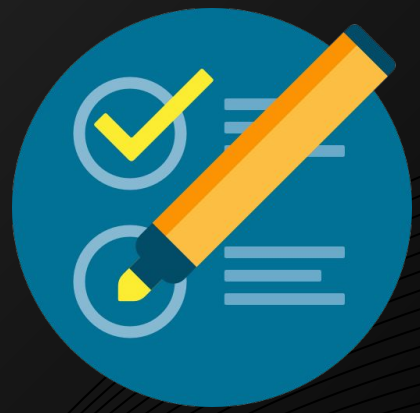
Web page
1st Design

Car
information

Alerts

- What will be done in next sprint?

Connection car-website



Approval tests

Web development

Demonstration

Car Dashboard

Explanation

Smoke Detection

Recap from the last sprint

Smoke Detection

Explanation

Raspberry PI Software Architecture Improvement

Connect and reconnect the USB communication between the Raspberry Pi and the STM32

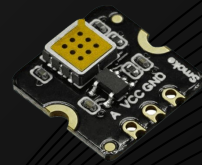
Verify the successful resumption of communication

Smoke Detection

Explanation

Integration of Smoke Detectors

- Integration of Sensors into the Software and Hardware Architecture



It works for dry steam but not for wet steam

...

How to test this?

...

Burning paper with a lighter (Dangerous?)

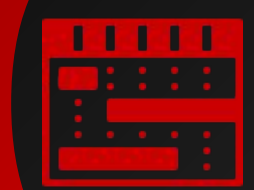
Smoke Detection

Explanation

Improvements for the Next Sprint

- Find a solution to convert multiple sensors (currently done with different ADCs)
- Conduct experiments and create a measurement table
- Identify patterns for setting fire detection thresholds on infrared sensors and smoke detectors

03



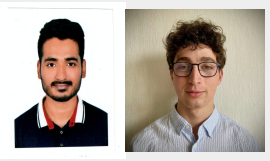
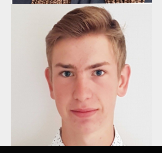
Sprint 4 planification

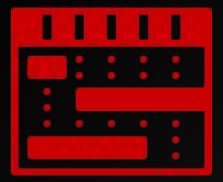
Next objectives

Team organisation

Organization for sprint 4

Sprint 4

01	User Interface	• Connection Car-Website • Communication User - Vehicle	
02	Fire Detection	• The car must be able to detect a fire	
03	Instrument reading	• Integration of AI model in ROS • Intruder detection with QR code	
04	Car navigation	• Implement SLAM for location • Create recording and replay mode for path planning	 



Intruder detection

Sprint 4

Story :

The car can detect non authorized employees within a building.
The car should trigger an alarm if a non authorized person is detected.

- State of the art communication ~2 days
- Develop the model ~2 days
- Setup of qr code reading functionality for employee identification ~3 days
- Perform the approval test and resolve bug ~4 days

Approval test :

- When an intruder is detected in the building, the an alarm is triggered.

Car Dashboard

Sprint 4

Story

As a user, I want the web page to connect with the car and see real-time information

- Connection Car <-> Webpage: Connect our user-friendly interface to the car.

70%

~9 days

- Manometer Reading: Show when a manometer is detected and show the value

90%

~4 days

- Filtering and Sorting: Include options for filtering or sorting the information based on user preferences or specific criteria

110%

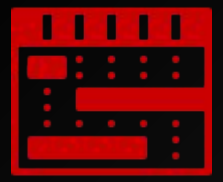
~2 days

Approval Tests

- All information is available and accessible
- Car values are updated within a maximum of 3 seconds after the change.

Demonstration

- The web page is functional, the user can navigate between the different data received



Fire detection

Sprint 4

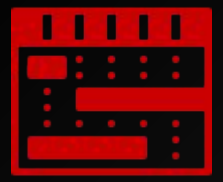
Story

The car must be able to detect fire along its path

- ➢ Implement a solution to convert multiple sensors ~2 days
- ➢ Verify each value converted by the ADC and compare the results against a reference ~2 days
- ➢ Conducting tests under actual conditions to validate smoke and infrared sensor ~5 days
- ➢ Establishing a threshold for each sensor, substantiated by test data ~4 days

Demonstration

- ➢ A video showing me simulating outdoor flames and triggering alarms
- ➢ Being able to use an ADC to convert my 4 analog sensors (simulating fire with a lighter).



Sprint 4

Car Location and Mapping

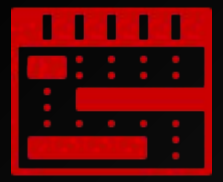
Story

As an user , I want to know exactly where the car is and the map of the environment it moving in.
Appropriate for places where external signals like GPS are not readily available.

- Process Lidar data and IMU to simultaneously build a map of the environment
- Determine the car's position within that map
- Continuously update the car's estimated position as it moves

Approval test & Demonstration

- The map is shown on the virtual dashboard
- The position of the car is known on the map.



Sprint 4

Recording and replay mode

Story

As a user, I want to record a path that the car can redo exactly

- Create a mode to register a path using the joystick in a text file — ~7 days
- Read the text file to replay the path — ~5 days
- Add camera position control to read manometers — ~4 days (bonus)

Approval Tests

- The register and replay path are the same

Demonstration

- Play a recorded path inside the GEI corridor

Sprint 3

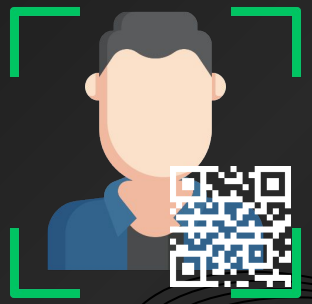
Next Demonstrations!

Path recording

The car plays a path previously recorded in the GEI corridor.

QR code recognition for intruder

The car recognizes intruders or unauthorized individuals on-site using the value from a QR code.

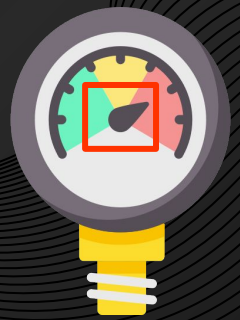


Virtual Dashboard

The website summarizes information and states of the car in **real-time**.

Read values from manometer

The can reads the value from manometer, to check if everything is fine



Alarm triggering & notification

Any abnormal activity should trigger the car's alarm and generate an alert on the website.

Indoor navigation

Technology

VigiWheels

Sensor

Smart

Communication

Autonomous

IoT

Agile

Team

Safety

Mobility

Actuator

Fire

Intruder

Future

Patrol

Your Thoughts, Please?