Forest Fire Fighter F³

2nd sprint review

Zilly team members :
Axel JORGE
Corentin TATGER
Liao ZHANG
Youssef AMARI
Ruiqi HU
Amelie MAIER

INSA TOULOUSE

TABLE OF CONTENTS

01 PLANNED AND REALIZED STORIES

Caption:

- Peanuts
- Simple
- Medium
- Hard
- Very Hard

Direction and movement

- Movement between 2 GPS locations
- Manage direction of front wheels

→ Ruiqi and Amélie

Computer vision

- Fire detection with machine learning

→ Corentin and Axel

Website and communication

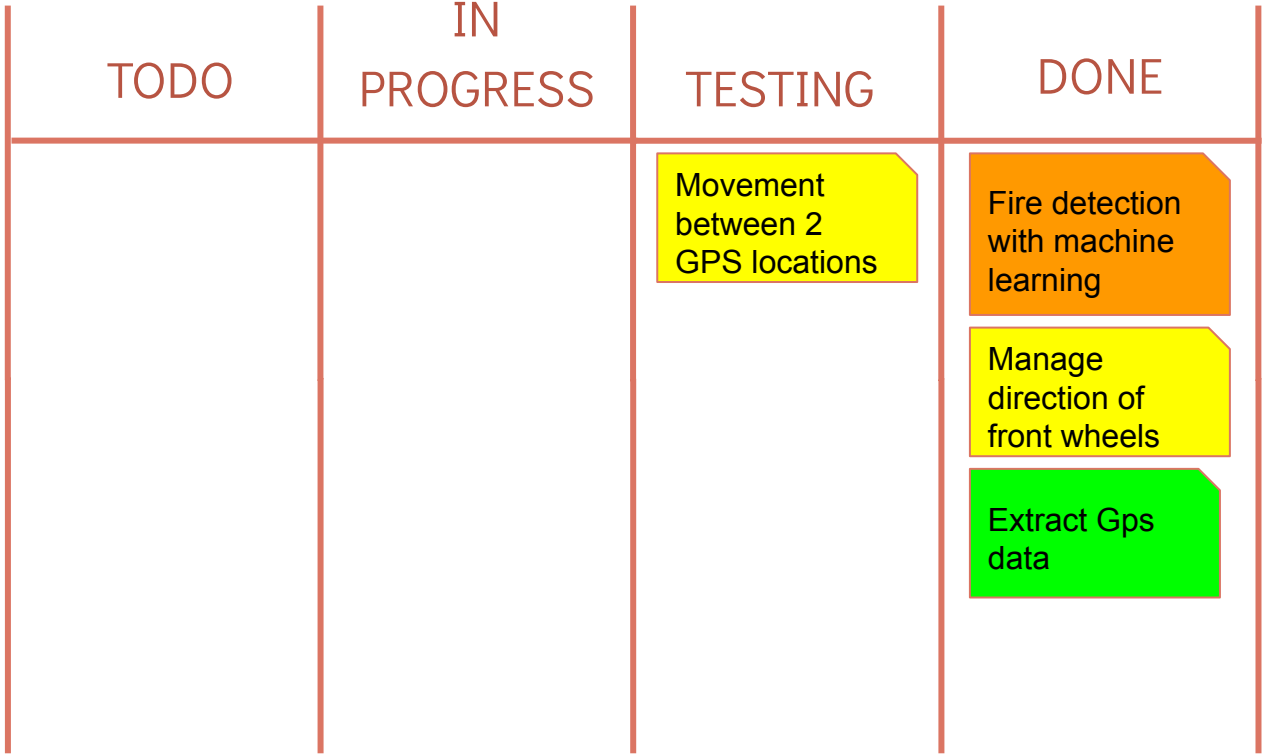
- Extract Gps data

→ Youssef and Liao

01 PLANNED AND REALIZED STORIES

Caption:

- Peanuts
- Simple
- Medium
- Hard
- Very Hard



02 DIRECTION AND MOVEMENT

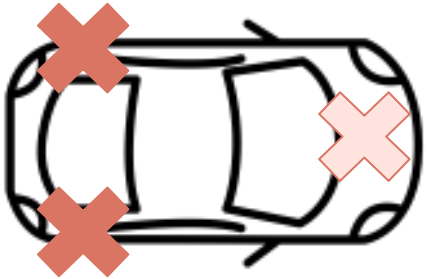
Stories

- User Story 1: Management of the direction of front wheels
- Test: Make a turn of 90 degrees on each sides

- User Story 2: Management of the movement of the car between two GPS location in straight lines
- Test: Going straight between two given GPS location and stop at it (50 cm accuracy)

02 DIRECTION AND MOVEMENT

Two motors at the back
-> Give power



One servomotor at the front
-> Give direction

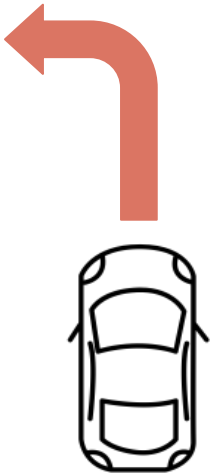
Command between 25 and 75
- If command < 50, go backward
- If command > 50, go forward

Command between 10 and 90
- If command < 50, turn left
- If command > 50, turn right

02 DIRECTION AND MOVEMENT

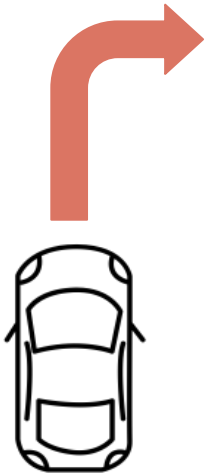
Tests

1st test : Turn left



Front command in [10, 40]

2nd test : Turn right



Front command in [60, 90]

02 DIRECTION AND MOVEMENT

Movement between two GPS location : How does it work ?

Without the GPS connected

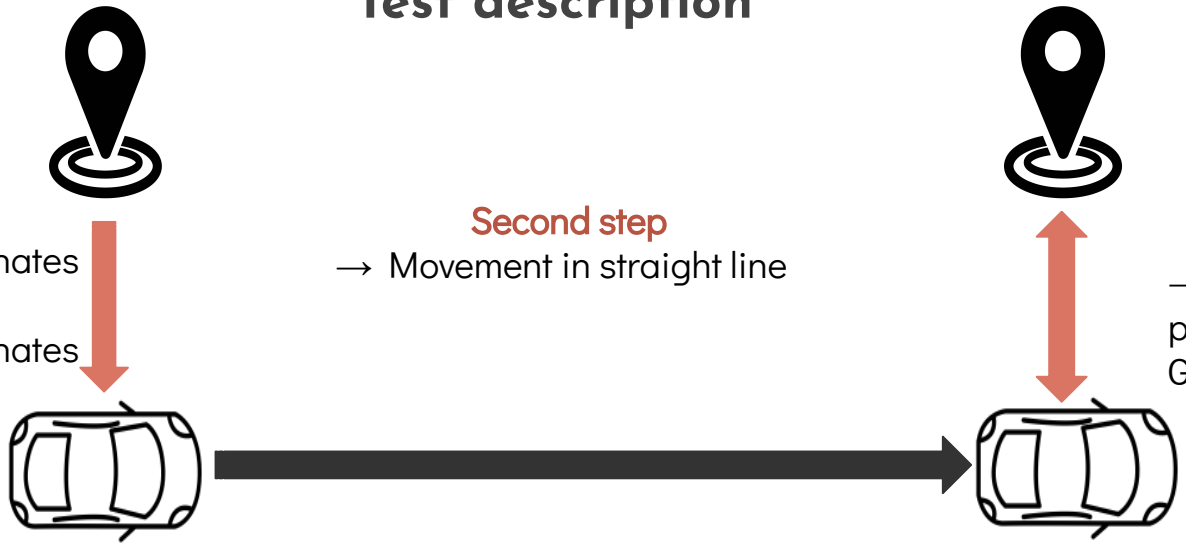
- **input =** two GPS location coordinates (latitude and longitude)
- **calculation of the distance** in meters between the two locations
- **management of the speed** of the car so that it joins the right position with 50 cm accuracy

With the GPS connected

→ not tested because of problems with the communication

02 DIRECTION AND MOVEMENT

Test description



First step
→ Give the GPS coordinates of the position of the car
→ Give the GPS coordinates of the location to join
→ Calculations

Second step
→ Movement in straight line

Third step
→ Verification of the position with the GPS

Acceptation threshold → the car stop at less than 50 cm of the location given

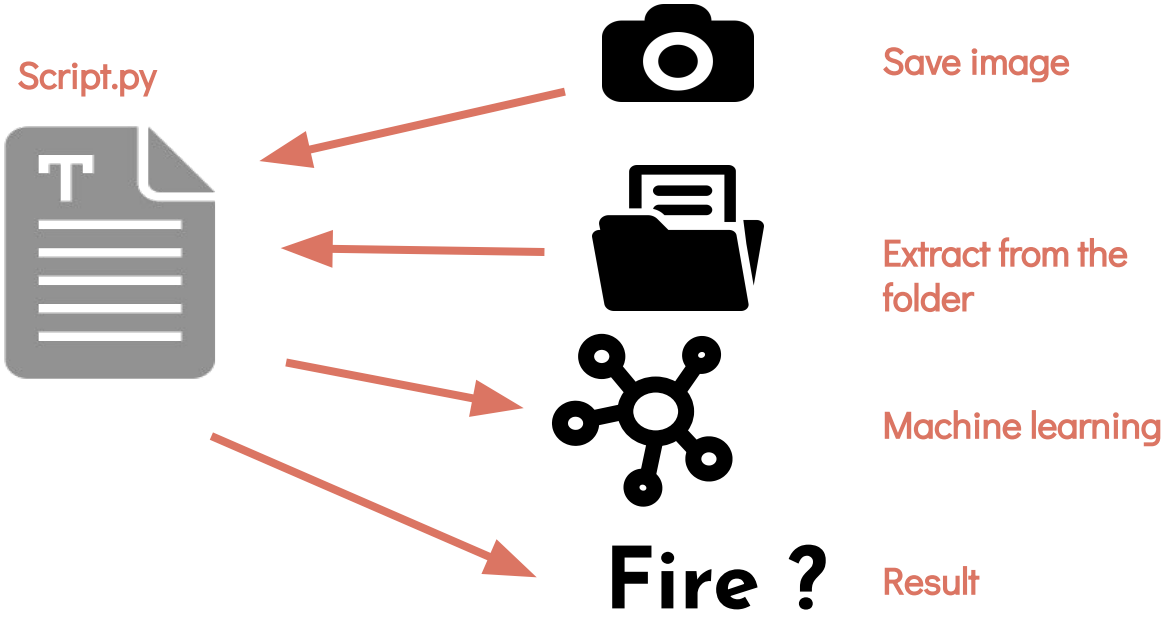
02 COMPUTER VISION

Stories

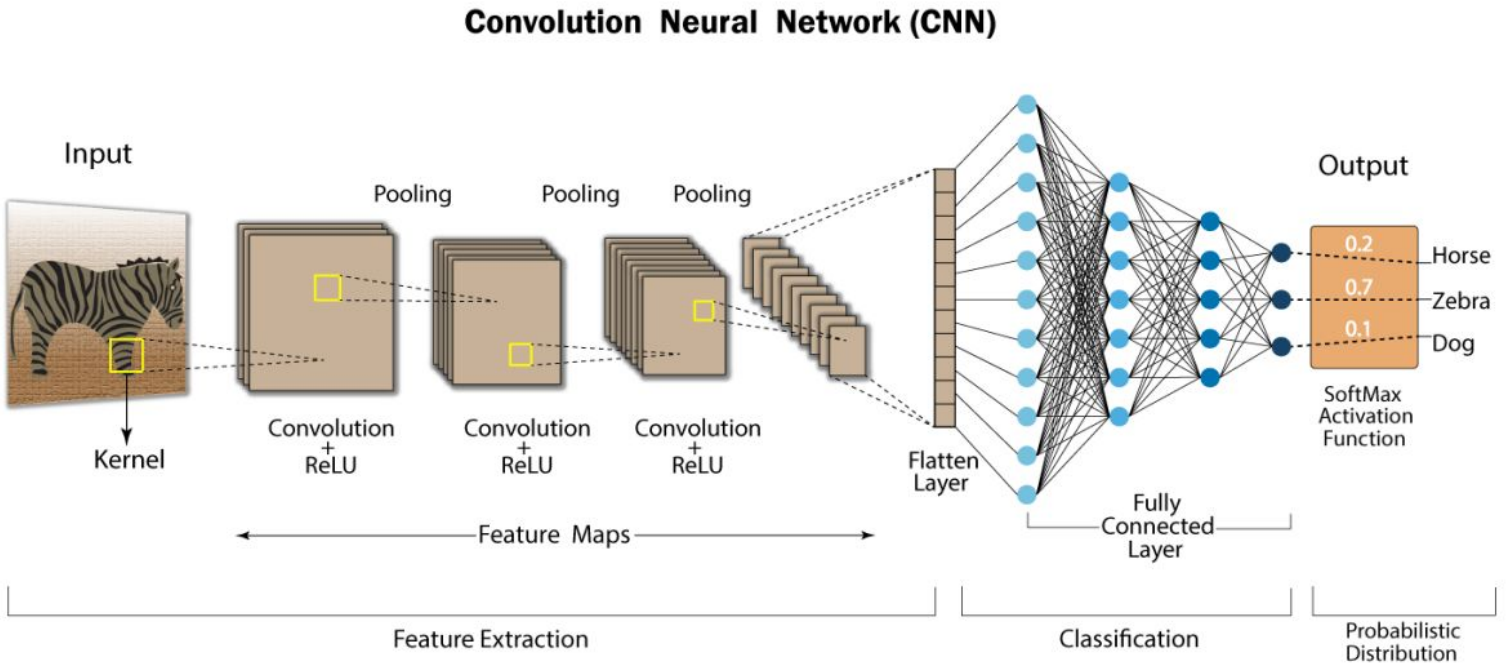
- User Story 1: The user should be able to check the presence of a fire
- Test: Capture image from the camera, save it and check for a fire outbreak. Repeat 10 times. Done when the program is right for 9 out of 10 images

02 COMPUTER VISION

Camera test



02 COMPUTER VISION



02 COMPUTER VISION

```
        frame = cv2.resize(frame, None, fx=0.5, fy=0.5, interpolation=cv2.INTER_AREA)
        cv2.imshow('Input', frame)

        c = cv2.waitKey(1)
        if c == 27:
            break

    _,Frame=cap.read()
    Frame = cv2.resize(Frame, dim,interpolation=cv2.INTER_AREA)
    cv2.imwrite(File_name,Frame)

transform = transforms.Compose( # composing several transforms together
    [transforms.ToTensor(), # to tensor object
    transforms.Normalize((0.5, 0.5, 0.5), (0.5, 0.5, 0.5))]) # mean = 0.5, std = 0.5

```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
Segmentation fault (core dumped)
corentin@corentin-GE70-2PE:~/Desktop/TSFlow$ bin/python3 /home/corentin/Desktop/TSFlow/Test_jetson.py
bash: bin/python3: No such file or directory
corentin@corentin-GE70-2PE:~/Desktop/TSFlow$ /bin/python3 /home/corentin/Desktop/TSFlow/Test_jetson.py
Frame_0.jpg
^[^[^[^[^CTraceback (most recent call last):
  File "/home/corentin/Desktop/TSFlow/Test_jetson.py", line 60, in <module>
    Get_data()
  File "/home/corentin/Desktop/TSFlow/Test_jetson.py", line 35, in Get_data
    ret, frame = cap.read()
KeyboardInterrupt

corentin@corentin-GE70-2PE:~/Desktop/TSFlow$ /bin/python3 /home/corentin/Desktop/TSFlow/Test_jetson.py
Frame_0.jpg
['ImageSet/Fire', 'ImageSet/No_Fire']
No_Fire
corentin@corentin-GE70-2PE:~/Desktop/TSFlow$ /bin/python3 /home/corentin/Desktop/TSFlow/Test_jetson.py
Frame_0.jpg
['ImageSet/Fire', 'ImageSet/No_Fire']
Fire
corentin@corentin-GE70-2PE:~/Desktop/TSFlow$ /bin/python3 /home/corentin/Desktop/TSFlow/Test_jetson.py
```

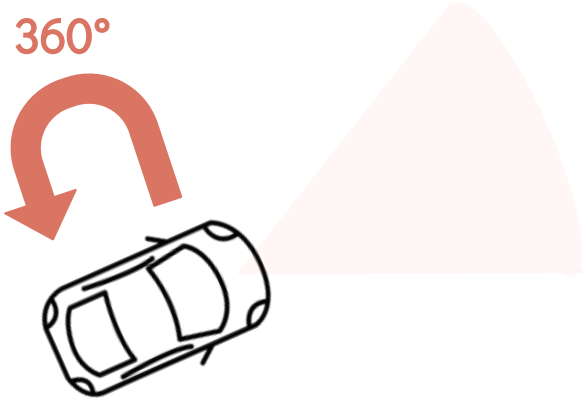
02 COMPUTER VISION

Test Results :

Variable	Expected value	Result
Global Accuracy	90%	92%
Rate of false Negatives	5%	4.2%
Rate of false Positives	0%	3.8%

02 COMPUTER VISION

Next improvements



OBJECTIVES

- 360° checking
- 1 image processed per seconde
- Send an image to the website when a fire is detected
- ~0% of false positives
- -5% of false negative

02 WEBSITE AND COMMUNICATION

Gps module/Data processing

Stories

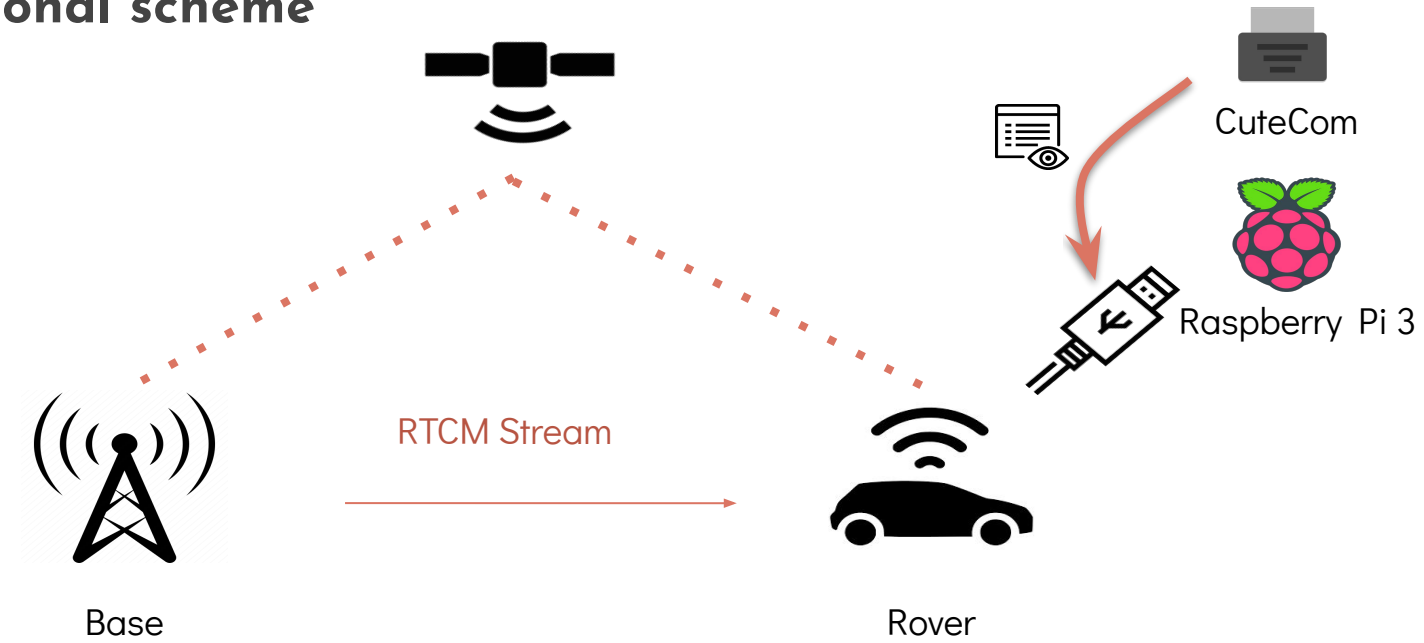
- User Story 1: Exportation and acquisition of vehicle location information
- Test: Observation of the data received through the GNSS module and processed data (vehicle coordinates) in real-time in the Raspberry Pi.

Done, when the precise coordinates (within 5 m) are achieved and comparison between displayed coordinates and actual location pretty match.

02 WEBSITE AND COMMUNICATION

Gps module/Data processing

Operational scheme



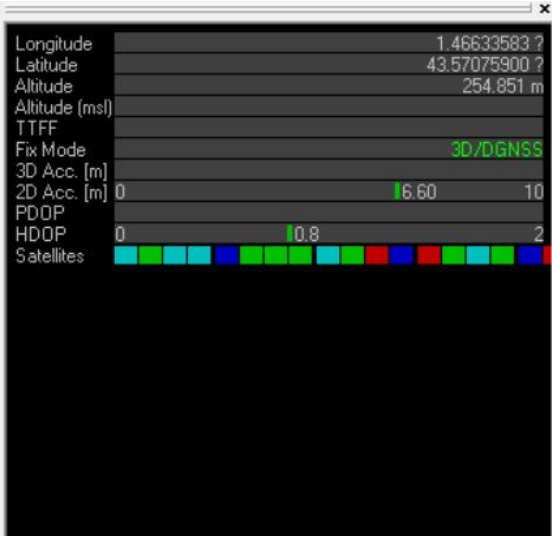
02 WEBSITE AND COMMUNICATION

Gps module/Data processing

Positioning Data (U-Centre)

NMEA - PUBX - 00 (Position Data) 2857

Parameter	Value	Unit	Description
UTC	161538.00	hhmmss.sss	Universal time coordinated
Lat	4334.24554	ddmm.mmmm	Latitude
Northing Indicator	N		N=North, S=South
Lon	00127.98015	dddmm.mmmm	Longitude
Easting Indicator	E		E=East, W=West
Alt (HAE)	254.851	m	Altitude (above ellipsoid)
Status	D3		NF=No Fix, ...
Horizontal Accuracy	6.6	m	Horizontal accuracy
Vertical Accuracy	11	m	Vertical accuracy
SOG	0.234	km/h	Speed Over Ground
COG (true)	25.91	deg	Course Over Ground (true)
VD	-0.188	m/s	Velocity Down
Age of DGPS Corr	0.0	s	Age of Differential Corrections
HDOP	0.82		Horizontal Dillution of Precision
VDOP	1.38		Vertical Dillution of Precision
TDOP	0.92		Time Dillution of Precision
SVs Used	8		Number of SVs used for Navigation
DR Status	0		Dead Reckon Status Flags

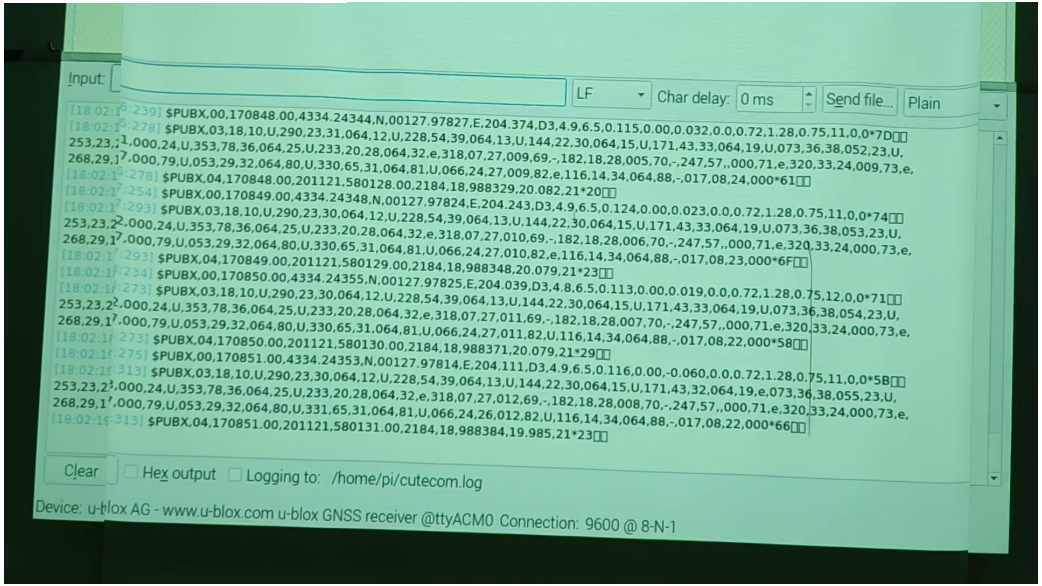


02 WEBSITE AND COMMUNICATION

Gps module/Data processing

Data observed (CuteCom)

- Position data (00)
- Satellite data (03)
- Time/date data (04)



02 WEBSITE AND COMMUNICATION

Data flow

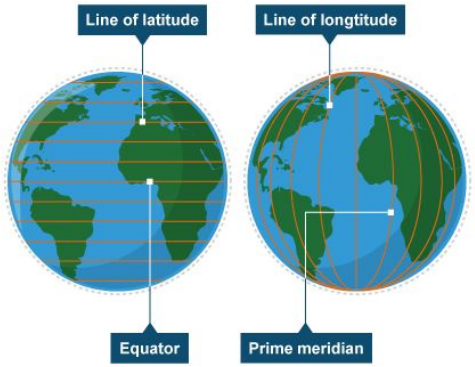
Gps module/Data processing

[20:03:08:230] $PUBX,00,190327.00,4334.25783,N,00127.97922,E,213.608,D3,11,13,0.341,235.28,-0.166,0.0,0.98,1.57,1.04,10,0,0*55

Time UTC
hhmmss.ss

Latitude
ddmm.mm

Longitude
dddmm.mm



Caption:
h: Hour
m: Minute
s: Second
d: Degree

02 WEBSITE AND COMMUNICATION

Gps module/Data processing

Data processed

- Parser in real time
- Sorted data
- [time, latitude, longitude]

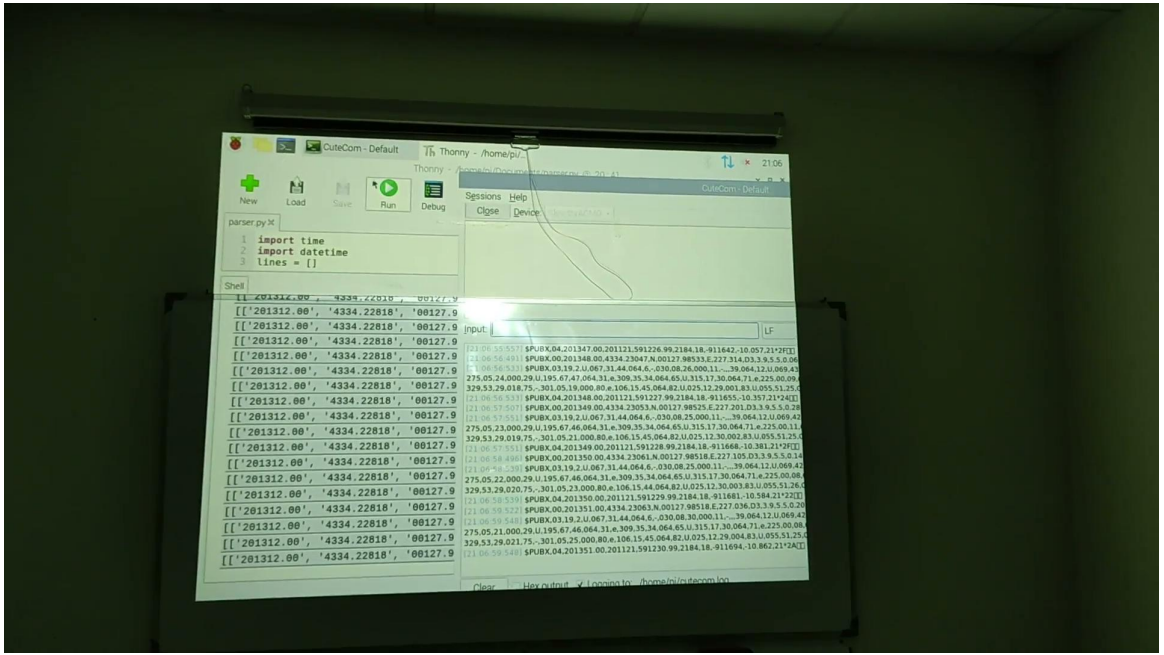
Python (Thonny)

02 WEBSITE AND COMMUNICATION

Gps module/Data processing

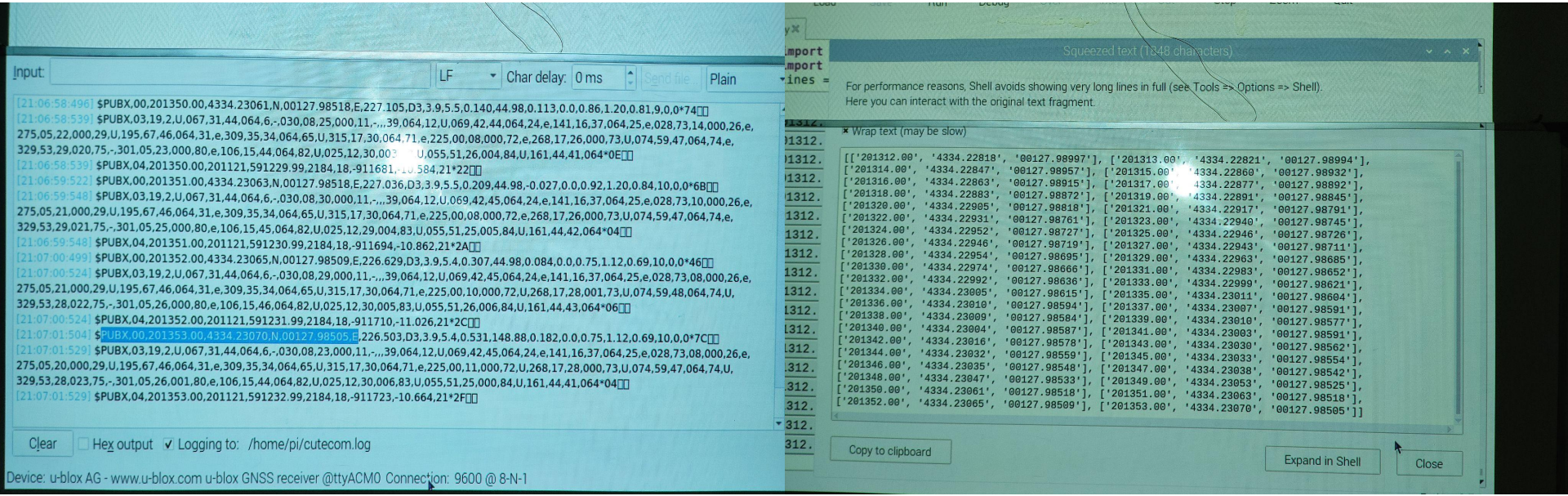
Data processed

- CuteCom on the right
- CuteCom writes slowly -> Empty lists at beginning
- Python parser (Real time)
  - Sleeping for 1 s
- Keeping useful data

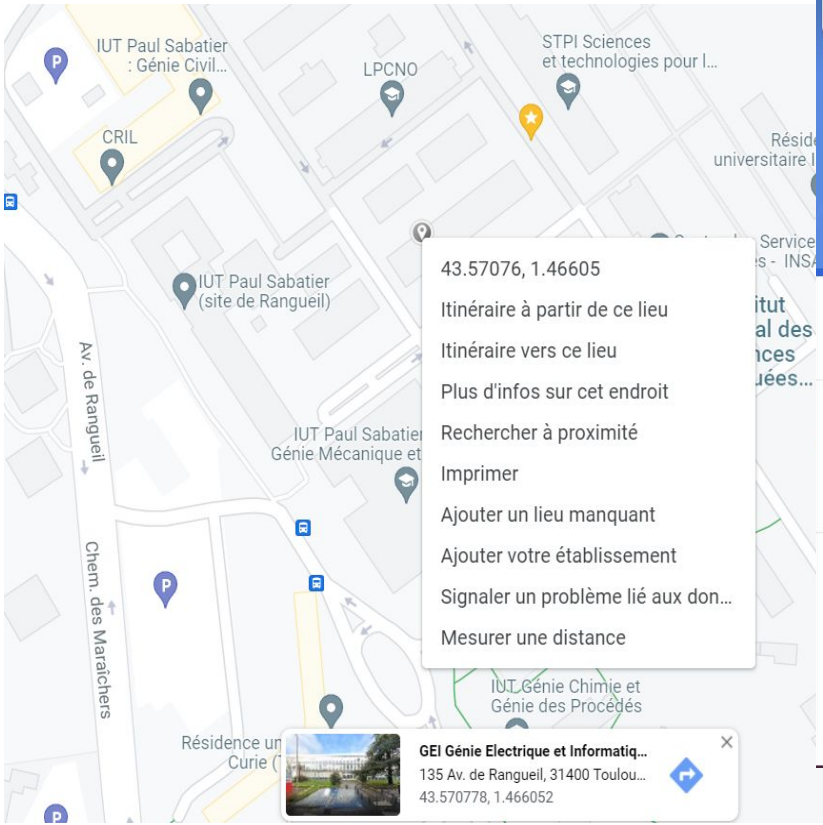


02 WEBSITE AND COMMUNICATION

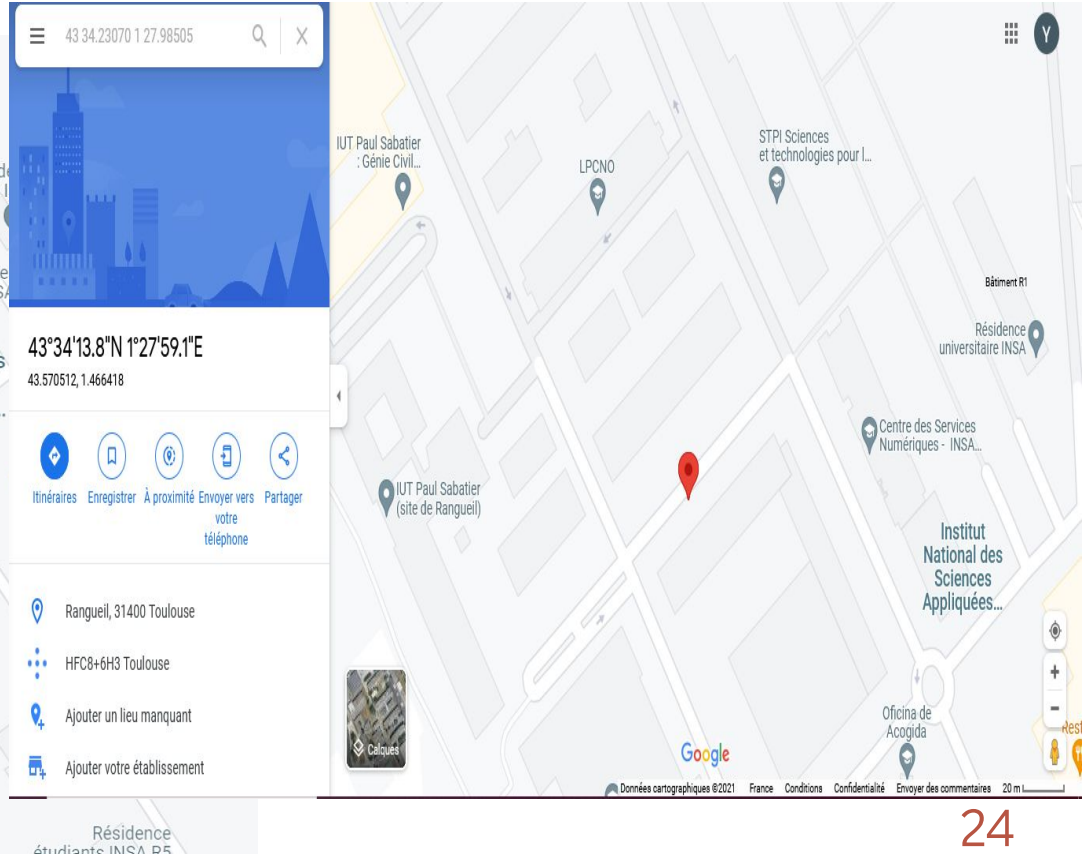
Gps module/Data processing



Actual coordinates



GPS coordinates



02 WEBSITE AND COMMUNICATION

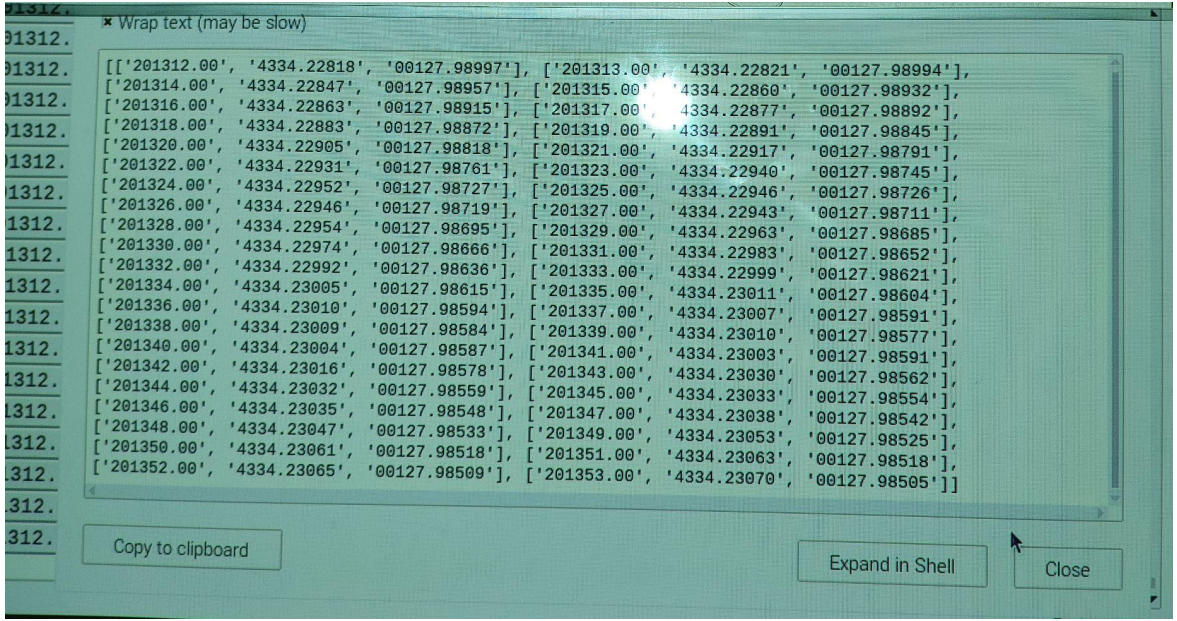
Gps module/Data processing

Data processed

- 41 seconds -> 41 lists of data
- Real coordinates of GEI at INSA Toulouse: -> **latitude = 43°57076”**
  -> **longitude = 1°46605”**

Average % of error for latitude = **0.524 %**

Average % of error for longitude = **12.7 %**



03 NEXT SPRINT ?

DIRECTION AND MOVEMENT

- User Story 1: **Direction and movement between three GPS location with real time management**
- Test: **Given three GPS location, the car is able to join each location (50 cm accuracy)**

WEBSITE AND COMMUNICATION

- User Story 2: **Coordinates and images sent and displayed on website and can be used by motors**
- Test: **Coordinates can be viewed on website in real time and can be used thanks to MQTT**

COMPUTER VISION

- User Story 3: **Fire detection process**
- Test: **~0% rate of false positives, -5% rate of false negatives**

04 CONCLUSION

Thank you for your attention

Any questions ?