電動車即時資料監控顯示平台(Realtime Sensor Display)

## 確認開發規格

根據金屬中心提供文件:

<https://hackmd.io/@cocobird231/Hk5geSc7i>

* **系統環境**
  + OS: Ubuntu 22.04 or Ubuntu 20.04 (x64)
  + RAM: 16GB or higher
  + WLAN: 1Gbps or higher
* **ROS2 Distro**
  + \*Humble (Ubuntu 22.04)
  + Foxy (Ubuntu 20.04)

\*: recommended

* **ROS2 Installation**
  + Humble installation guide: [docs.ros.org](https://docs.ros.org/en/humble/Installation/Ubuntu-Install-Debians.html)
  + Foxy installation guide: [docs.ros.org](https://docs.ros.org/en/foxy/Installation/Ubuntu-Install-Debians.html)
    - If using foxy distro, after installation completed, highly recommended replace parameter.py from [here](https://github.com/ros2/rclpy/blob/humble/rclpy/rclpy/parameter.py) (humble distro).
    - The default parameter.py location: /opt/ros/foxy/lib/python3.8/site-packages/rclpy/parameter.py
* **ROS2 Tutorials (Humble Distro)**
  + \*Topic (one publisher and multiple subscribers)
    - C++: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Cpp-Publisher-And-Subscriber.html)
    - Python3: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Py-Publisher-And-Subscriber.html)
  + Service (one server and multiple clients)
    - C++: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Cpp-Service-And-Client.html)
    - Python3: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Writing-A-Simple-Py-Service-And-Client.html)
  + Custom Interfaces
    - Create interfaces: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Custom-ROS2-Interfaces.html)
    - Implement custom interfaces: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Beginner-Client-Libraries/Single-Package-Define-And-Use-Interface.html)
  + ROS2 Launch File
    - Create launch file: [docs.ros.org](https://docs.ros.org/en/humble/Tutorials/Intermediate/Launch/Creating-Launch-Files.html)

\*: Commonly used for sensor message

## 籌備程式專案及開發環境

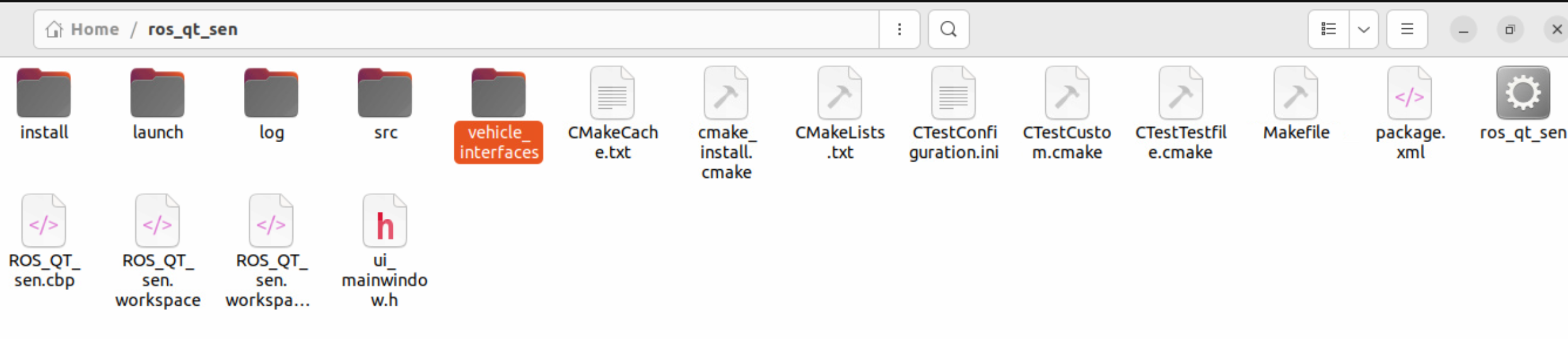
於Linux cli 輸入:

source /opt/ros/humble/setup.bash

新增資料夾以作為工作目錄

mkdir -p ~/ros\_qt\_sen/src

將vehicle\_interfaces放入工作目錄



設定自己的CMakeLists與pakage.xml

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

## 介面顯示與設計

一張含有 文字 的圖片

自動產生的描述

## 程式開發與設計

建立測試節點以利測試

一張含有 文字 的圖片

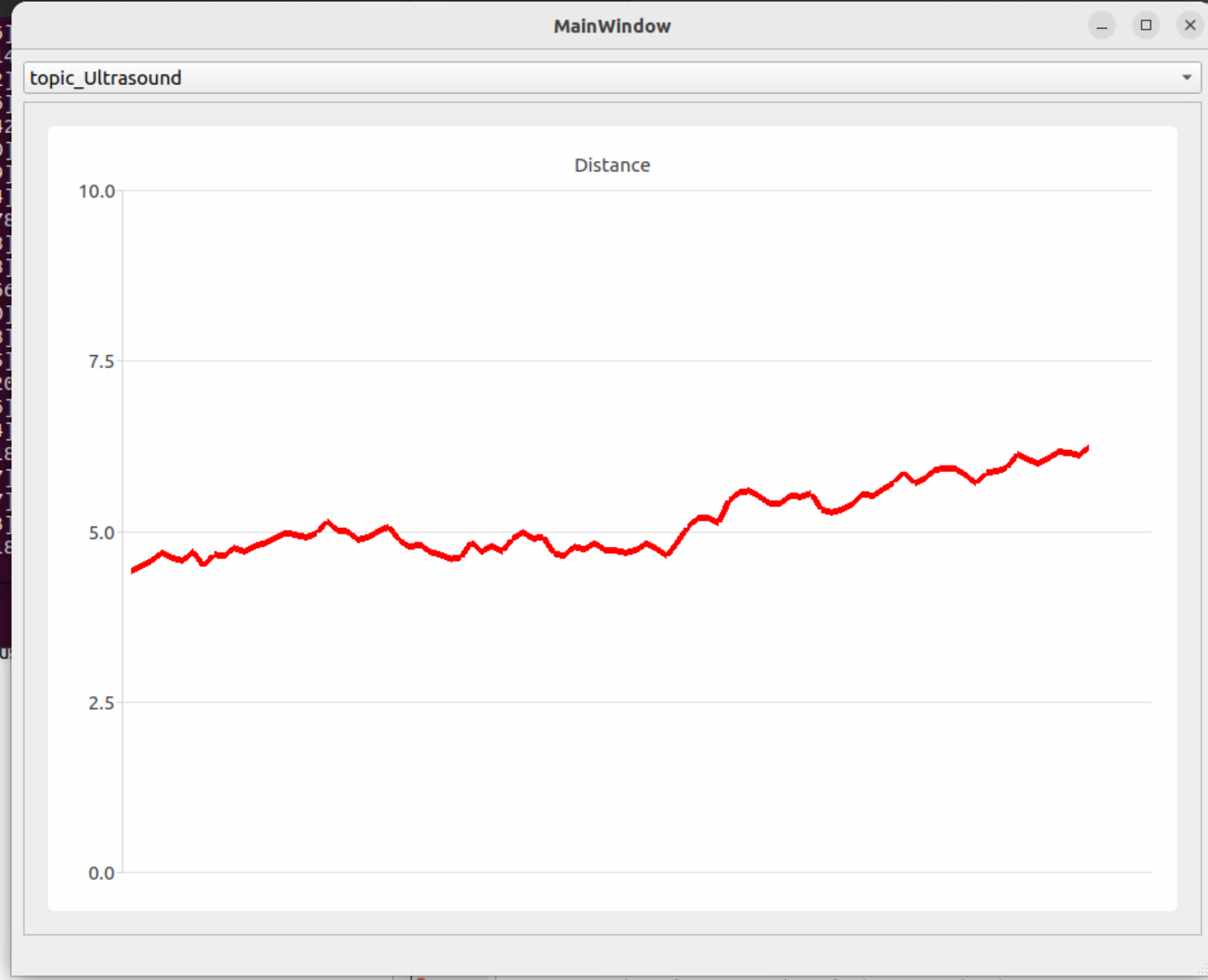
自動產生的描述

撰寫介面功能

一張含有 文字 的圖片

自動產生的描述

功能測試



## 安裝與測試專案

安裝Linux Github desktop

sudo wget <https://github.com/shiftkey/desktop/releases/download/release-3.1.1-linux1/GitHubDesktop-linux-3.1.1-linux1.deb>

sudo apt-get install gdebi-core

sudo gdebi GitHubDesktop-linux-3.1.1-linux1.deb

將專案clone下來



安裝ROS2

可參考網址: <https://docs.ros.org/en/humble/Installation/Ubuntu-Install-Debians.html>

輸入CLI設定安裝環境:

locale # check for UTF-8

sudo apt update && sudo apt install locales

sudo locale-gen en\_US en\_US.UTF-8

sudo update-locale LC\_ALL=en\_US.UTF-8 LANG=en\_US.UTF-8

export LANG=en\_US.UTF-8

locale # verify settings

sudo apt install software-properties-common

sudo add-apt-repository universe

sudo apt update && sudo apt install curl -y

sudo curl -sSL [https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -o /usr/share/keyrings/ros-archive-keyring.gpg](https://raw.githubusercontent.com/ros/rosdistro/master/ros.key%20-o%20/usr/share/keyrings/ros-archive-keyring.gpg)

echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/ros-archive-keyring.gpg] http://packages.ros.org/ros2/ubuntu $(. /etc/os-release && echo $UBUNTU\_CODENAME) main" | sudo tee /etc/apt/sources.list.d/ros2.list > /dev/null

安裝CLI:

sudo apt update

sudo apt upgrade

sudo apt install ros-humble-desktop

sudo apt install ros-humble-ros-base

sudo apt install ros-dev-tools

安裝QT5 -charts:

sudo apt install libqt5charts5-dev

設定CLI環境變數:

source /opt/ros/humble/setup.bash

進入專案編譯

編譯vehicle\_interfaces

cd vehicle\_interfaces

colcon build --packages-select vehicle\_interfaces

設定vehicle\_interfaces環境

. ./install/setup.sh

cd ..

編譯ros\_qt\_sen

colcon build --packages-select ros\_qt\_sen

執行程式:

./build/ros\_qt\_sen/ros\_qt\_sen

執行結果:

一張含有 文字 的圖片

自動產生的描述