# Embedded Systems Development

## Task 5.2C RPi - Making GUI

### Hardware Required

Raspberry Pi with Raspbian installed (NOOBS)
Keyboard, mouse and monitor (HDMI) to plug into the Raspberry Pi. LEDs, Breadboard, jumper wires

### Software Required

An IDE for Python development.

### Pre-requisites: You must do the following before this task

1) RPi tasks 4.1P and 5.1P
2) Research on how to develop GUIs for Python. Popular packages to develop GUIs are Qt, Tkinter or Kivy (there might be others, and you are free to choose the library you are most comfortable with). You can find many good tutorials on Youtube. Examples are as below:
    a) Qt https://youtu.be/Vde5SH8e1OQ
    b) Tkinter https://www.youtube.com/watch?v=ap-ABFNCBoE
    c) Kivy https://www.youtube.com/watch?v=B79miUFD_ss

### Task Objective

In this task, you will create simple GUI interfaces with Raspberry Pi. RPi supports python, which has many GUI packages including Qt, TkInter and Kivy. As before, while this task is focusing on Python, you can use any programming language you are most comfortable with (and supported by RPi platform) to perform this task and future tasks incorporating a RPi. For example, if you are using C++, Qt has a complete GUI package for C++ which is very similar to the Python package.

In this task, based on what you learnt from the video tutorials, you are required to build a simple GUI interface to turn on one of three LED lights. Given that this is a credit task, you are required to do a fair amount of research and upskilling to be able to perform the following steps. So, plan in advance.

#### Steps:

1) Build a simple circuit board with RPi that enables you to turn on 3 LEDs separately. Name them red, green, and blue.

2) Create a GUI that has the following: 3 radio buttons that correspond to the 3 LED, and a button to exit the GUI.
3) Create code for the GUI so that: when you click on each of the radio button, the corresponding LED will turn on, and all others are off.

## Task Submission Details

Q1: Submit a video that demonstrates the system working.

Q2: Create a repository named SIT210-Task5.2C-RPiGUI on Github. Upload your code to the repository. Include the link to your repository here.

Q3: Describe a real-life usage scenario for your system.

*Remember, anytime you submit a task to OnTrack, it is a good practice to check the status of any existing tasks, and the future tasks you are expected to complete. If you have got feedback on previous tasks, you may need to fix and resubmit some of your work. You want to check out why, so that you can learn from this and make it faster and easier to accomplish later work to the required standard.*