

Figure 3.9: Interface organization

### 3.3.2 Message Created

The message created consists on the variables that the user can control (intensity and location) and the number of motors that are active. To this message name it was given the name *GUIDados.msg*.

```

uint8 numberOfMotors
uint8[ ] intensity
uint8[ ] location

```

Figure 3.10: Message GUIDados

Figure 3.10 illustrates the message that was created. The type of variable *uint8* (unsigned 8-bit int) was used due to the fact of being the smallest variable that ROS messages works with. It is the equivalent of the variable type *char* in ROS. While the number of motors is a simple number, both the intensity and location are an array of numbers. Since 16 motors are being used, the size of the array is 16.

### 3.3.3 Package Organization

Figure 3.11 shows how the package *qtgui* is organized.

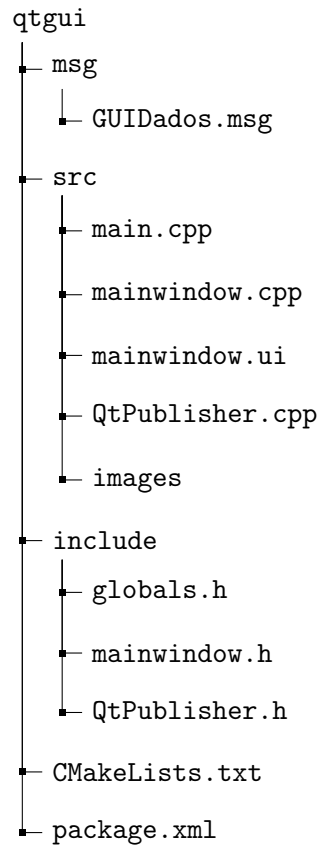


Figure 3.11: Organization of the *qtgui* package

### 3.3.4 CMakeLists

ROS is built using *catkin\_make* and Qt with *qmake*. The first thing done was to change the built option from *qmake* to *cmake*, so it would be easier to mix the Qt CMakeLists and the ROS CMakeLists.

Since ROS is being used in the interface node, some important things such as the *catkin\_libraries* and *QT\_libraries* had to be added in the CMakeLists file. Important packages like *Qt4* and *qt\_ros\_interface* were also added.

### 3.3.5 User Interface

The interface offers the user two working modes:

- Motor control.
- Pattern.