Configure QTrobot TTS language

OVERVIEW

Level: *Intermediate*
Goal: *learn about how to set a language of QTrobot TTS*
Requirements:

- Quick start with coding on QTrobot
- Create a ROS python project
- QTrobot interfaces using ROS Services

NOTE

This tutorial is for the customers how bought additional TTS languages!

In this tutorial you will learn how to set a language for **QTrobot Speech interface**.

Create a python project

First we create a python project for our tutorial. let's call it `tutorial_qt_speech` and add the required python file:

```
cd ~/catkin_ws/src
catkin_create_pkg tutorial_qt_speech rospy roscpp -D "Set QTrobot TTS
language"
cd tutorial_qt_speech/src
touch tutorial_qt_speech_node.py
chmod +x tutorial_qt_speech_node.py
```

Code

Now lets see how we can change a voice (language) of QTrobot. Following are some standard supported languages:

- **en-US** (American English)
- **fr-FR** (French)
- **de-DE** (German)

You may have different languages installed on your QTrobot. This tutorial will use English and French language.

Open the `tutorial_qt_speech_node.py` file and add the following code:

```
#!/usr/bin/env python
import sys
```

```
import rospy
from std_msgs.msg import String
from qt_robot_interface.srv import *

if __name__ == '__main__':
    rospy.init_node('my_tutorial_node')
    rospy.loginfo("my_tutorial_node started!")

    # define a ros service
    speechConfig = rospy.ServiceProxy('/qt_robot/speech/config',
speech_config)

    # define a ros service
    speechSay = rospy.ServiceProxy('/qt_robot/speech/say', speech_say)

    # block/wait for ros service
    rospy.wait_for_service('/qt_robot/speech/say')

    # block/wait for ros service
    rospy.wait_for_service('/qt_robot/speech/config')

    try:
        status = speechConfig("en-US",0,0)
        if status:
            speechSay("Hello, I am speaking English")
            status = False
        rospy.sleep(1)
        status = speechConfig("fr-FR",0,0)
        if status:
            speechSay("Bonjour, Je parle français")


    except KeyboardInterrupt:
        pass

    rospy.loginfo("finsihed!")
```

Explanation

ROS Services are defined by srv files, which contains a request message and a response message. First we import all from `qt_robot_interface.srv`. This will import all srv files that are under `qt_robot_interface.srv`. We need to use `speech_config`.

TIP

How do we know which service an interface uses? well, There is a useful command in ROS which tells you that:

```
rosservice info /qt_robot/speech/config
Type: qt_robot_interface/speech_config
Args: language pitch speed
```

```
...
```

Then we defined a service `/qt_robot/speech/config` and call `rospy.wait_for_service()` to block until a service is available. Finally we called a ROS service with a wanted language and if everything is ok service will return "True". After that we call `/qt_robot/speech/say` service with text message to check the configured language.