TurtleSim

Aldin Cebo

I. Src code

CPP code for **Hello.h**:

```
#include <iostream>
#include <ros/ros.h>
#include <template_package/JoystickValue.h>
#include <geometry_msgs/Twist.h>
class Hello{
private:
    ros::NodeHandle nh;
    ros::Publisher pub;
    ros::Subscriber sub;
    void callback(const template_package::
    JoystickValue& msg);
public:
    Hello();
    ~Hello();
    void run();
};
```

CPP code for **Hello.cpp**:

```
#include <hello_world/Hello.h>
Hello::Hello(){
  pub = nh.advertise<geometry_msgs::Twist>
  ("joystick_value", 1);
  sub = nh.subscribe("/turtle1/cmd_vel",
  1, &Hello::callback, this);
}
Hello::~Hello(){
}
void Hello::callback(const template_package::JoystickValue& msg){
  geometry_msgs::Twist vel_msg;
  int x = msg.x;
  int y = msg.y;
  if (x > 500 && x < 550 && y > 500 && y < 550) {//Middle
    vel_msg.linear.x = 0;
    vel_msg.angular.z = 0;
  } else if (x > 500 && x < 550 && y < 300) {//Up
    vel_msg.linear.x = 2;
  } else if (x > 500 && x < 550 && y > 600) {//Down
    vel_msg.linear.x = -2;
  } else if (x > 600 && y > 500 && y < 550) {//Left
    vel_msg.angular.z = 1;
  } else if (x < 300 && y > 500 && y < 550) {//Right
    vel_msg.angular.z = -1;
  }
  pub.publish(vel_msg);
}

void Hello::run(){
  ros::spin();
}
```

CPP code for **hello_world.cpp**:

```
#include <hello_world/Hello.h>
int main (int argc, char** argv) {
    ros::init(argc, argv, "turtle_joystick");
    Hello hello;
    hello.run();
}
```