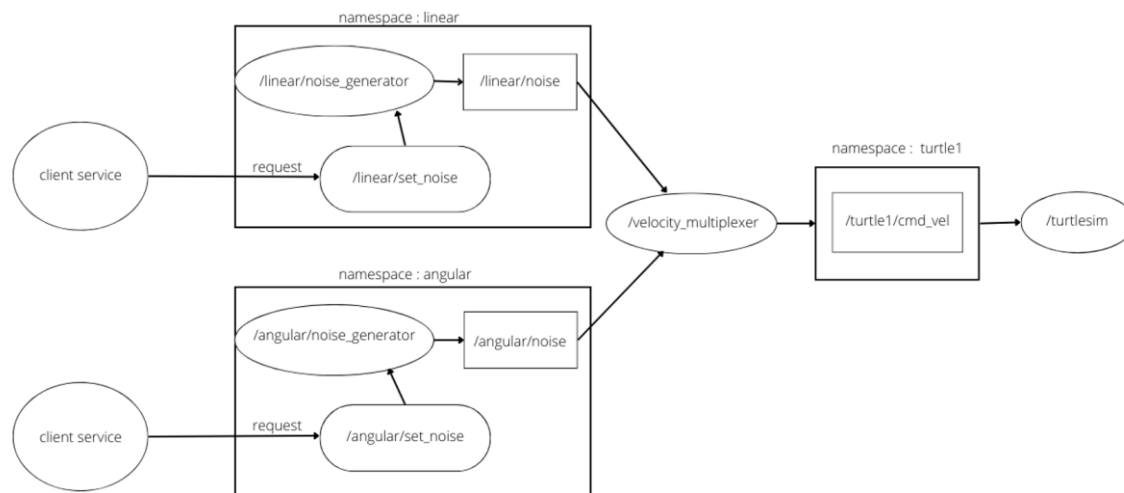


## System Architecture



## ขั้นตอนการทำงาน

### 1. การสร้าง Package และการแก้ไข CMakeLists

โดยในการสร้าง Package จะใช้ script สำหรับการสร้าง package อัตโนมัติจาก

[https://github.com/tchoopojcharoen/ROS2\\_pkg\\_cpp\\_py](https://github.com/tchoopojcharoen/ROS2_pkg_cpp_py)

โดยการใช้คำสั่ง `. /install_pkg.bash fra333_ws fra333_lab1_20` ในการสร้างแพคเกจที่ชื่อ `fra333_lab1_20` โดยมี Workspace ชื่อ `fra333_ws` จากนั้นนำ package ที่ถูกสร้างไปแทนที่ในโฟลเดอร์ `lab1` สำหรับการเปลี่ยนชื่อแพคเกจเดิมของ `lab1` จะทำให้ได้แพคเกจใหม่ที่มีชื่อ `fra333_lab1_20` และมีไฟล์ script [ `Noise_gernerator.py` , `velocity_mux.py` , `random_walk.launch.py` ] ภายในแพคเกจ `fra333_lab1_20`

ภายใน Workspace ก็จะมีอีกหนึ่งแพคเกจที่มีชื่อว่า `lab1_interface` เป็นแพคเกจสำหรับเก็บไฟล์ `.srv` เพื่อใช้สำหรับการเขียน service

การที่จะทำให้ไฟล์ Python สามารถรันได้โดยผ่าน ROS2 จำเป็นที่จะต้องใส่ชื่อไฟล์ executable ที่จะต้องกำหนดภายใน CMakeLists ซึ่งจะได้ดังนี้

```
# Install Python executables
install(PROGRAMS
  scripts/noise_generator.py
  scripts/velocity_mux.py
  DESTINATION lib/${PROJECT_NAME}
)
```

และการระบุโฟลเดอร์ที่จำเป็นสำหรับการใช้เรียกตัว Launch file ของ ROS2

```
install(DIRECTORY
  # add directories here
  launch
  DESTINATION share/${PROJECT_NAME})
```

## 2. การสร้าง Node Publisher และ Service [ Noise\_generator.py ]

การสร้างกำหนดชื่อ Node ผ่านคำสั่ง `super().__init__(" ")`

```
super().__init__("noise_generator")
```

2.1 การสร้าง Publisher สำหรับการส่งข้อมูล Noise ผ่าน Topic /noise และมีชนิดข้อมูลที่ส่งเป็น Standard Message แบบ Float64 โดยที่ข้อมูลของ Float64 จะประกอบไปด้วยตัวแปร data ที่ใช้สำหรับการเก็บค่า

### std\_msgs/Float64 Message

**File:** `std_msgs/Float64.msg`

#### **Raw Message Definition**

```
float64 data
```

หลังจากนั้นทำการสร้าง Timer สำหรับการส่ง publish ข้อมูลโดยที่จะส่งในอัตราส่วน 1:rate โดยที่มีค่า ปกติอยู่ที่ 5.0 Hz หรือ rate = 5.0

```
self.noise_publisher = self.create_publisher(Float64, '/noise', 10)
self.timer = self.create_timer(1/self.rate, self.timer_callback)
```

โดยจะเรียกฟังก์ชัน timer\_callback สำหรับการ publish ข้อมูล Noise ส่งไปยัง Topic /noise

```
def timer_callback(self):
    noise = Float64()
    noise.data = np.random.normal(self.mean, self.variance**0.5)
    self.noise_publisher.publish(noise)
```

ในการสร้างสัญญาณรบกวนให้เป็นแบบ Gaussian Distribution โดยใช้ library numpy ผ่านฟังก์ชัน random.normal( mean , Standard Derivation) และนำค่าสัญญาณที่ได้ นำไป publish ไปยัง Topic /noise ที่ได้ตั้งไว้ตอนเริ่มสร้าง publisher

## 2.2 การสร้าง Service

```
self.set_noise_service = self.create_service(SetNoise, '/set_noise', self.set_noise_callback)
```

ใช้การสร้าง Server สำหรับ service โดยมีชื่อ set\_noise โดยที่เมื่อเรียก service จะทำการเรียกฟังก์ชัน set\_noise\_callback สำหรับใช้ในการเปลี่ยนค่า mean และ variance ผ่าน Request ของ Service

```
def set_noise_callback(self, request: SetNoise.Request, response: SetNoise.Response):
    self.mean = request.mean.data
    self.variance = request.variance.data
    self.get_logger().info(f'request\n mean: {self.mean} variance: {self.variance}')
```

เราสามารถกำหนดค่า Type ของ Message ที่ใช้ในการรับค่าที่กำหนด Request และ Response ผ่านไฟล์ Setnoise.srv ที่อยู่ภายใต้แพ็คเกจ lab1\_interfaces

```
from lab1_interfaces.srv import SetNoise
```

โดยค่า Request ที่รับจะเป็น std\_msgs/Float64 mean และ variance ซึ่งสามารถเขียนในไฟล์ Setnoise.srv ได้ดังนี้

```
lab1_interfaces > srv > ≡ SetNoise.srv
1   std_msgs/Float64 mean
2   std_msgs/Float64 variance
3   ---
```

3. การสร้าง Subscriber สำหรับการรับข้อมูลสัญญาณรบกวน และ การสร้าง Publisher ในการส่งไปยัง turtlesim

#### 3.1 การสร้าง Subscriber สำหรับการรับสัญญาณรบกวน

```
self.angular_subscription = self.create_subscription(Float64, '/angular/noise', self.angular_vel_sub_callback, 10)
self.linear_subscription = self.create_subscription(Float64, '/linear/noise', self.linear_vel_sub_callback, 10)
```

กำหนดประเภทของข้อมูล Message ที่จะรับซึ่งก็คือ Float64 โดยผ่าน Topic /angular/noise และ /linear/noise โดยที่ข้อมูลที่ได้รับจะเข้าผ่านฟังก์ชัน

angular\_vel\_sub\_callback และ linear\_vel\_sub\_callback ใช้สำหรับการเปลี่ยนค่า cmd\_vel ในการส่งค่าสัญญาณรบกวนไปยัง turtlesim

```
def linear_vel_sub_callback(self, msg: Float64):
    self.cmd_vel.linear.x = msg.data

def angular_vel_sub_callback(self, msg: Float64):
    self.cmd_vel.angular.z = msg.data
```

#### 3.2 การสร้าง publisher สำหรับการส่งสัญญาณรบกวนไปยัง turtlesim

```
self.cmd_vel = Twist()
self.cmd_pub = self.create_publisher(Twist, '/turtle1/cmd_vel', 10)
```

สร้างตัวแปร cmd\_vel โดยกำหนดให้เป็น Message ประเภท Twist ซึ่งจะประกอบไปด้วยตัวแปร linear และ angular ที่เป็นตัวแปรชนิด Vector3 โดยที่ตัว publisher จะส่งข้อมูลไปยัง Topic /turtle1/cmd\_vel โดยที่กำหนดให้ชนิดของ Message ที่จะส่งเป็น Twist

```
self.timer = self.create_timer(1/self.rate,self.timer_callback)
```

โดยเราจะกำหนด timer ให้มีค่าความถี่ในการส่งคือ  $1/\text{rate}$  โดยที่มีค่า Default คือ  $\text{rate} = 5$  หรือ  $5.0$  Hz

ในฟังก์ชัน `timer_callback` ก็จะเป็นการสั่งให้ตัว `publisher` ส่งข้อมูลตัวแปร `cmd_vel` ที่เป็นข้อมูลแบบ Twist ไปยัง Topic ที่ได้กำหนดไว้ ( `/turtle1/cmd_vel` )

```
def timer_callback(self):  
    self.cmd_pub.publish(self.cmd_vel)
```

#### 4. การสร้าง Launch file [ `random_walk.launch.py` ]

##### 4.1 การสร้างกำหนด argument

กำหนดสร้างตัวแปร `rate` เพื่อใช้ในการกำหนดค่า `rate` สำหรับการปรับความถี่ในการส่งข้อมูล โดยที่กำหนดให้มีค่า Default อยู่ที่  $5.0$  และใช้คำสั่ง `add_action` เพื่อเป็นการเพิ่ม argument ในการปรับค่า `rate` สำหรับการ launch file หรือ Node ที่ต้องการ

```
rate = LaunchConfiguration('rate')  
rate_launch_arg = DeclareLaunchArgument('rate',default_value='5.0')  
launch_description.add_action(rate_launch_arg)
```

##### 4.2 การสร้าง Launch Description สำหรับการรัน Node ที่ต้องการ

โดยที่ Node แรกจะเป็น Node ที่ใช้สำหรับการส่งสัญญาณรบกวนที่เป็นแบบ linear โดยจะใช้ executable ที่อยู่ภายใต้แพ็คเกจ `fra333_lab1_20` ที่มีชื่อว่า `noise_generator.py` โดยที่จะมี namespace คือ 'linear' ซึ่งจะทำให้ Topic ที่ส่งออกไปยัง Node นี้จะมีชื่อ `/linear` นำหน้าอยู่เสมอทำให้จากเดิม ที่ `noise_generator.py` จะส่งออก Topic `/noise` กลายเป็น `/linear/noise` และ service จาก `/set_noise` กลายเป็น `/linear/set_noise` โดยที่จะรับ argument `rate` เพิ่มเข้ามาสำหรับการปรับค่าความถี่ในการส่ง และในการ remapping เนื่องจากในไฟล์ `noise_generator.py` จะส่งข้อมูลไปยัง Topic `/noise` ซึ่งเมื่อใส่ namespace จะทำให้ Topic กลายเป็น `/linear//noise` ทำให้ข้อมูลไม่สามารถส่งไปยัง Topic ที่ถูกต้องได้จึงใช้ remapping จาก `'/noise'` ให้กลายเป็น `'noise'`

จะทำให้ Topic ที่ได้กลายเป็น /linear/noise และ /linear/set\_noise ตามที่ต้องการ

```
linear = Node(  
    package='fra333_lab1_20',  
    executable='noise_generator.py',  
    namespace= 'linear',  
    arguments=[rate],  
    remappings=[  
        ('/noise', 'noise'),  
        ('/set_noise', 'set_noise')  
    ]  
)  
launch_description.add_action(linear)
```

จากนั้นทำเช่นเดียวกันกับ Node ที่ต้องการสร้างสัญญาณรบกวนสำหรับ Angular

```
angular = Node(  
    package='fra333_lab1_20',  
    executable='noise_generator.py',  
    namespace= 'angular',  
    arguments=[rate],  
    remappings=[  
        ('/noise', 'noise'),  
        ('/set_noise', 'set_noise')  
    ]  
)  
launch_description.add_action(angular)
```

ทำการเพิ่ม Node Velocity\_mux สำหรับการรับค่าสัญญาณรบกวนและส่งไปยัง turtlesim โดยที่มี argument เช่นเดียวกับกับ noise\_generator ในการกำหนดค่าความถี่ในการส่งข้อมูล

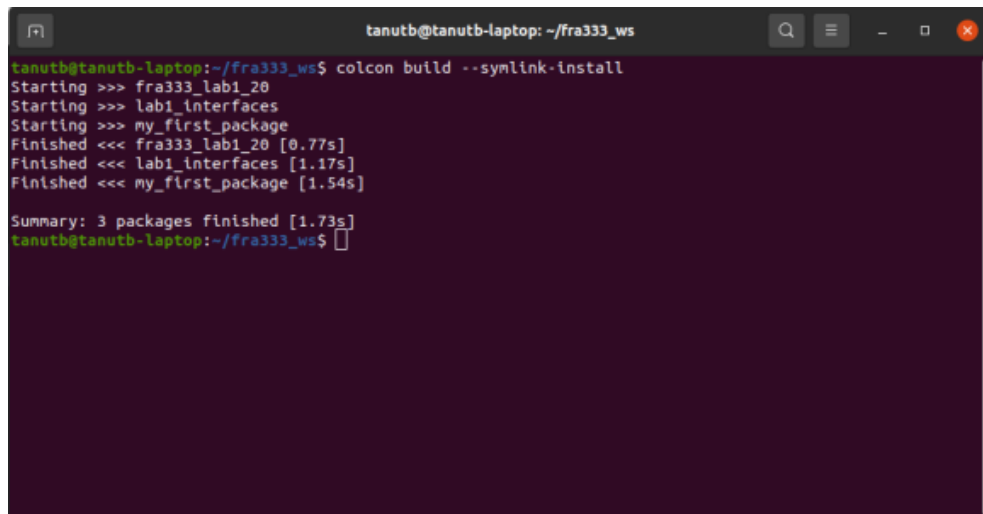
```
Vmux = Node(  
    package='fra333_lab1_20',  
    executable='velocity_mux.py',  
    namespace= '',  
    arguments=[rate]  
)  
launch_description.add_action(Vmux)
```

เพิ่ม Node turtlesim\_node สำหรับการรันให้ turtlesim รับค่าที่ส่งไปจาก Velocity\_mux

```
turtlesim_node = Node(  
    package='turtlesim',  
    executable='turtlesim_node',  
)  
launch_description.add_action(turtlesim_node)
```

และหลังจากที่กำหนด Node ที่ต้องการจะรัน ใช้คำสั่ง add\_action สำหรับการเพิ่ม Node ที่เราได้กำหนดไว้ไปยัง Description ของ Launch file จะทำให้เมื่อเราเรียกใช้ launch file จะเรียกเปิดใช้ Node ทั้งหมดที่เรากำหนดและตั้งค่า Node ที่เปิดตามที่ต้องการ

หลังจากที่เขียน Node และ Launch file ทั้งหมดเรียบร้อยแล้วจึงทำการ ใช้คำสั่ง Colcon build ภายใน workspace เพื่อเป็นการ build package ที่เขียนขึ้น

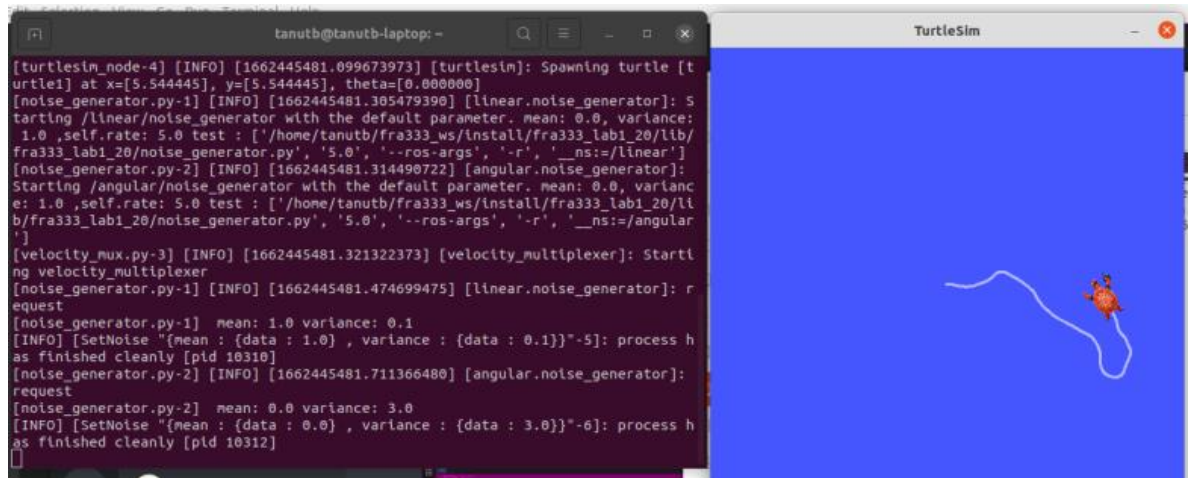
A terminal window titled 'tanutb@tanutb-laptop: ~/fra333\_ws' showing the output of the 'colcon build --symlink-install' command. The output lists the starting and finishing times for three packages: fra333\_lab1\_20, lab1\_interfaces, and my\_first\_package. The summary indicates that 3 packages were finished in 1.73s.

```
tanutb@tanutb-laptop:~/fra333_ws$ colcon build --symlink-install  
Starting >>> fra333_lab1_20  
Starting >>> lab1_interfaces  
Starting >>> my_first_package  
Finished <<< fra333_lab1_20 [0.77s]  
Finished <<< lab1_interfaces [1.17s]  
Finished <<< my_first_package [1.54s]  
  
Summary: 3 packages finished [1.73s]  
tanutb@tanutb-laptop:~/fra333_ws$
```

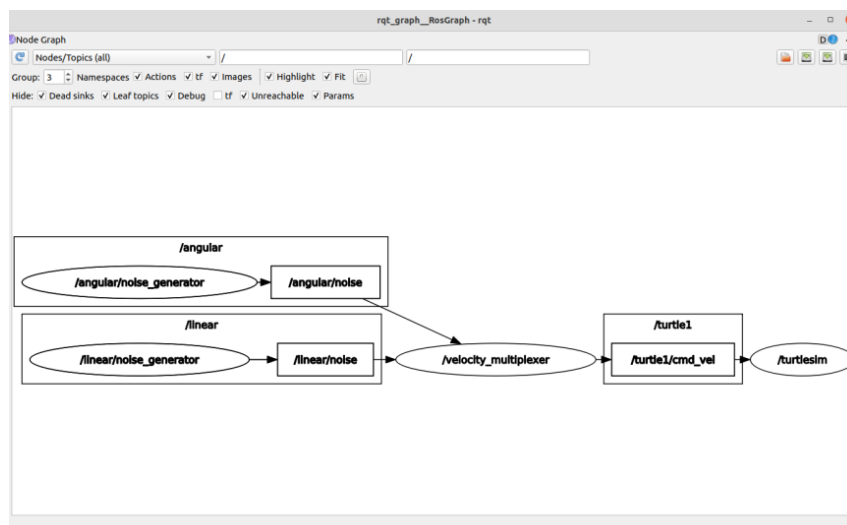
ผลสรุป

ทดสอบการเรียกใช้ Node ผ่าน Launch file

การเรียก Launch file โดยใช้คำสั่ง `ros2 launch fra333_lab1_20 random_walk.launch.py`



โดยเมื่อเรียกใช้ Launch file พบว่า Node ทุก Node สามารถรันผ่าน Launch file ได้ตามปกติและมี turtlesim ขึ้นตามที่ต้องการจากนั้นตรวจสอบ Topic ของ ros2 ที่มีอยู่หลังจากเปิดใช้งาน launch file เพื่อตรวจสอบความถูกต้องของ Topic และการส่ง Message ผ่าน rqt\_graph





จากนั้นตรวจสอบ Server Service ผ่านคำสั่ง `ros2 service list`

```
tanutb@tanutb-laptop:~$ ros2 service list
/angular/noise_generator/describe_parameters
/angular/noise_generator/get_parameter_types
/angular/noise_generator/get_parameters
/angular/noise_generator/list_parameters
/angular/noise_generator/set_parameters
/angular/noise_generator/set_parameters_atomically
/angular/set_noise
/clear
/kill
/linear/noise_generator/describe_parameters
/linear/noise_generator/get_parameter_types
/linear/noise_generator/get_parameters
/linear/noise_generator/list_parameters
/linear/noise_generator/set_parameters
/linear/noise_generator/set_parameters_atomically
/linear/set_noise
/reset
/spawn
/turtle1/set_pen
/turtle1/teleport_absolute
/turtle1/teleport_relative
/turtlesim/describe_parameters
/turtlesim/get_parameter_types
/turtlesim/get_parameters
```

จาก Terminal จะเห็นได้ว่ามี Server Service ที่ชื่อ `/linear/set_noise` และ `/angular/set_noise` ตามที่ต้องการ

**ทดสอบการใช้ launch file ผ่าน argument ในการกำหนด rate หรือความถี่ในการส่งข้อมูล**

ทดสอบโดยการเรียกใช้ launch file โดยมี argument คือ rate โดยปรับให้ rate มีค่าอยู่ที่ 40

โดยใช้คำสั่ง `ros2 launch fra333_lab1_20 random_walk.launch.py rate:=40`

```
tanutb@tanutb-laptop: ~  
tanutb@tanutb-laptop:~$ ros2 launch fra333_lab1_20 random_walk.launch.py rate:=40
```

```
INFO] [SetNoise "{mean : {data : 1.0} , variance : {data : 0.1}}"-5]: process started with pid [10897]  
INFO] [SetNoise "{mean : {data : 0.0} , variance : {data : 3.0}}"-6]: process started with pid [10898]  
turtlesim_node-4] [INFO] [1662445613.573128906] [turtlesim]: Starting turtlesim with node name /turtlesim  
turtlesim_node-4] [INFO] [1662445613.576298118] [turtlesim]: Spawning turtle [turtle1] at x=[5.544444  
], y=[5.544445], theta=[0.000000]  
noise_generator.py-1] [INFO] [1662445613.730685603] [linear.noise_generator]: Starting /linear/noise  
generator with the default parameter. mean: 0.0, variance: 1.0 ,self.rate: 40.0 test : ['/home/tanut  
b/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/noise_generator.py', '40', '--ros-args', '-r',  
__ns:=/linear']  
velocity_mux.py-3] [INFO] [1662445613.758049027] [velocity_muxer]: Starting velocity_muxer  
noise_generator.py-2] [INFO] [1662445613.761603580] [angular.noise_generator]: Starting /angular/noi  
se_generator with the default parameter. mean: 0.0, variance: 1.0 ,self.rate: 40.0 test : ['/home/tan  
utb/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/noise_generator.py', '40', '--ros-args', '-r'  
, '__ns:=/angular']  
noise_generator.py-1] [INFO] [1662445614.162927487] [linear.noise_generator]: request  
noise_generator.py-1] mean: 1.0 variance: 0.1  
INFO] [SetNoise "{mean : {data : 1.0} , variance : {data : 0.1}}"-5]: process has finished cleanly [p  
id 10897]  
noise_generator.py-2] [INFO] [1662445614.198938099] [angular.noise_generator]: request  
noise_generator.py-2] mean: 0.0 variance: 3.0  
INFO] [SetNoise "{mean : {data : 0.0} , variance : {data : 3.0}}"-6]: process has finished cleanly [p
```

จากการรัน command พบว่า การกำหนด rate เป็น argument ภายใน Launch file สามารถเปลี่ยน rate ภายใน Node ที่ใช้สำหรับการกำหนดการส่งค่าข้อมูลได้

## ทดสอบการเรียกใช้ Service

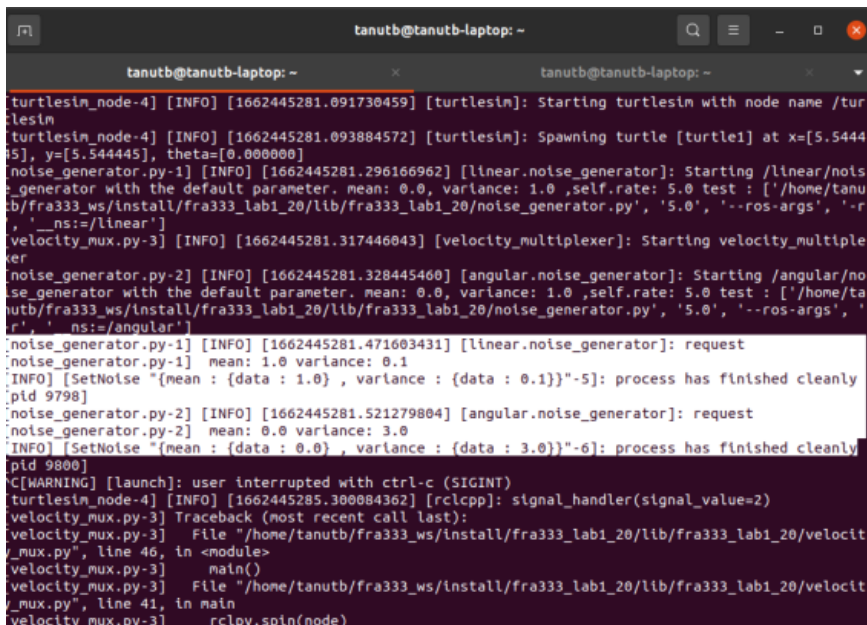
โดยการ command สำหรับการเรียก service ภายใน launch file สำหรับการทดสอบการเรียก Service ในการเปลี่ยนค่า mean และ variance ในขณะที่ Node ทั้งหมดเริ่มทำงาน

โดยที่จะกำหนดให้ Linear noise มีค่า mean และ variance อยู่ที่ 1.0 และ 0.1

Angular noise มีค่า mean และ variance อยู่ที่ 0.0 และ 3.0

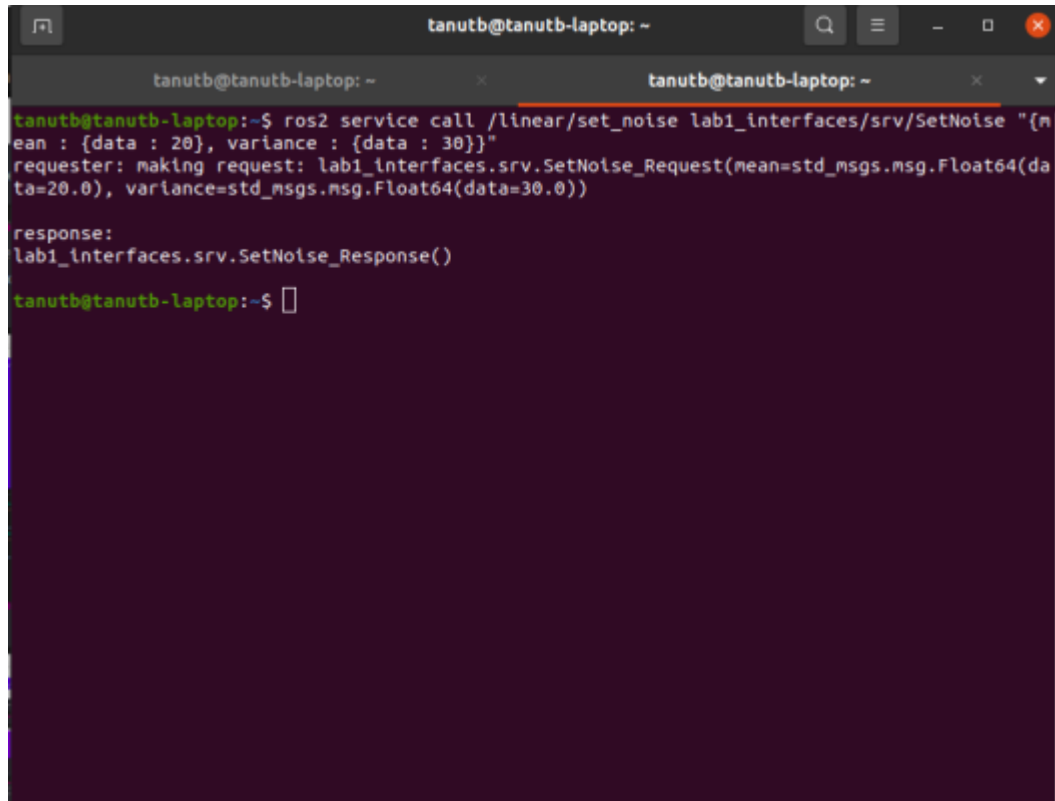
```
service_linear_set_noise = ExecuteProcess(  
    cmd = [[f'ros2 service call /linear/set_noise lab1_interfaces/srv/SetNoise "{(mean : {{data : 1.0}} , variance : {{data : 0.1}})}"',  
    shell=True  
    )  
    launch_description.add_action(service_linear_set_noise)  
    service_angular_set_noise = ExecuteProcess(  
        cmd = [[f'ros2 service call /angular/set_noise lab1_interfaces/srv/SetNoise "{(mean : {{data : 0.0}} , variance : {{data : 3.0}})}"',  
        shell=True  
        )  
    )
```

โดยเมื่อตรวจสอบจาก Terminal หลังจากเรียก Launch file พบว่า ได้มีการเรียกใช้ Service Setnoise ของทั้ง Linear และ Angular ผ่านทาง Launch file และได้ทำการเปลี่ยนค่าจากการเรียกใช้ service Setnoise ได้ตามที่ต้องการ



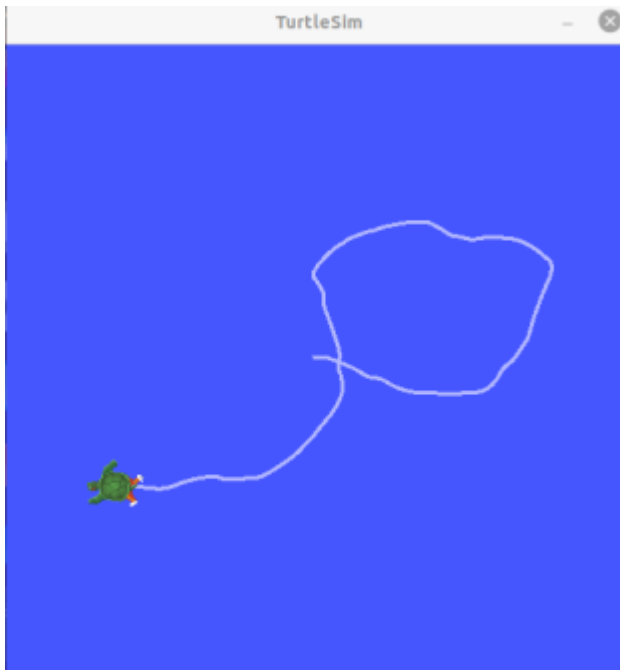
```
tanutb@tanutb-laptop: ~  
turtlesim_node-4] [INFO] [1662445281.091730459] [turtlesim]: Starting turtlesim with node name /tur  
lesim  
turtlesim_node-4] [INFO] [1662445281.093884572] [turtlesim]: Spawning turtle [turtle1] at x=[5.5444  
5], y=[5.544445], theta=[0.000000]  
noise_generator.py-1] [INFO] [1662445281.296166962] [linear.noise_generator]: Starting /linear/nois  
e_generator with the default parameter. mean: 0.0, variance: 1.0 ,self.rate: 5.0 test : ['/home/tanu  
tb/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/noise_generator.py', '5.0', '--ros-args', '-r  
, 'ns:=/linear']  
velocity_mux.py-3] [INFO] [1662445281.31746043] [velocity_mux.py-3]: Starting velocity_multiple  
xer  
noise_generator.py-2] [INFO] [1662445281.328445460] [angular.noise_generator]: Starting /angular/no  
ise_generator with the default parameter. mean: 0.0, variance: 1.0 ,self.rate: 5.0 test : ['/home/tanu  
tb/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/noise_generator.py', '5.0', '--ros-args', '-  
r', 'ns:=/angular']  
noise_generator.py-1] [INFO] [1662445281.471603431] [linear.noise_generator]: request  
noise_generator.py-1] mean: 1.0 variance: 0.1  
[INFO] [SetNoise "{(mean : {data : 1.0} , variance : {data : 0.1})}"-5]: process has finished cleanly  
pid 9798]  
noise_generator.py-2] [INFO] [1662445281.521279804] [angular.noise_generator]: request  
noise_generator.py-2] mean: 0.0 variance: 3.0  
[INFO] [SetNoise "{(mean : {data : 0.0} , variance : {data : 3.0})}"-6]: process has finished cleanly  
pid 9800]  
[WARNING] [launch]: user interrupted with ctrl-c (SIGINT)  
turtlesim_node-4] [INFO] [1662445285.300084362] [rclcpp]: signal_handler(signal_value=2)  
velocity_mux.py-3] Traceback (most recent call last):  
velocity_mux.py-3] File "/home/tanutb/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/velocit  
y_mux.py", line 46, in <module>  
velocity_mux.py-3]     main()  
velocity_mux.py-3] File "/home/tanutb/fra333_ws/install/fra333_lab1_20/lib/fra333_lab1_20/velocit  
y_mux.py", line 41, in main  
velocity_mux.py-3]     rclpy.spin(node)
```

การเรียกใช้ Service ผ่าน terminal



```
tanutb@tanutb-laptop: ~  
tanutb@tanutb-laptop: ~  
tanutb@tanutb-laptop:~$ ros2 service call /linear/set_noise lab1_interfaces/srv/SetNoise "{mean : {data : 20}, variance : {data : 30}}"  
requester: making request: lab1_interfaces.srv.SetNoise_Request(mean=std_msgs.msg.Float64(data=20.0), variance=std_msgs.msg.Float64(data=30.0))  
response:  
lab1_interfaces.srv.SetNoise_Response()  
tanutb@tanutb-laptop:~$
```

Turtlesim จากการรัน Node ทั้งหมด ผ่าน Launch file



Ros2 topic echo /turtle1/cmd\_vel

```
tanutb@tanutb-laptop: ~  
y: 0.0  
z: -0.4610976431186039  
---  
linear:  
  x: 0.7344953785245172  
  y: 0.0  
  z: 0.0  
angular:  
  x: 0.0  
  y: 0.0  
  z: 0.3623328190301011  
---  
linear:  
  x: 0.511587607192576  
  y: 0.0  
  z: 0.0  
angular:  
  x: 0.0  
  y: 0.0  
  z: 1.2472937643513125  
---  
linear:  
  x: 1.2339763956472292  
  y: 0.0  
  z: 0.0  
angular:  
  x: 0.0  
  y: 0.0  
  z: -2.222169263420258  
---
```

### สมาชิก

1. นาย ธนัท บำรุงวงศ์ศิริ 63340500026
2. นาย ภรภัทร พิชรพิมพ์พิสุทธิ 63340500026