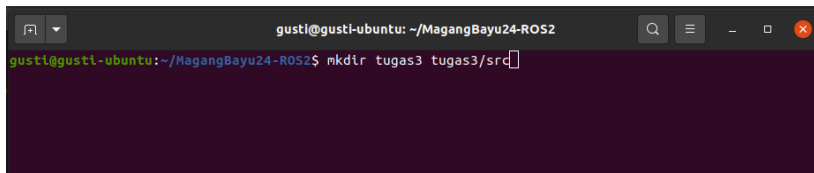


Nama : Gusti Gratia Delpiera

NRP : 5026231097

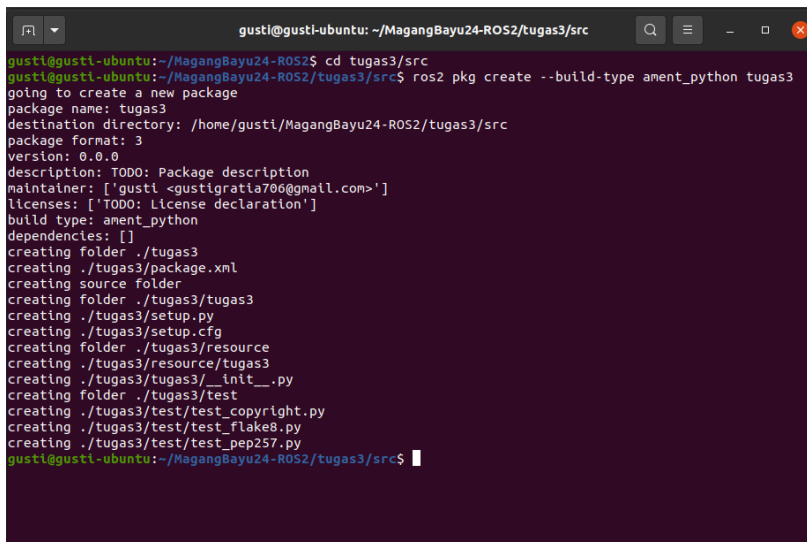
Laporan Penugasan 3

1. Langkah pertama setelah melakukan clone repository dengan menggunakan git clone, untuk mengerjakan penugasan ini adalah dengan membuat workspace, di sini saya membuat workspace bernama tugas3 dan juga direktori src



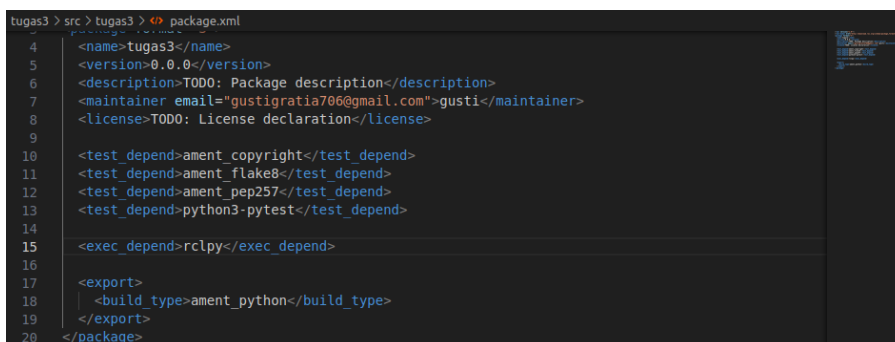
```
gusti@gusti-ubuntu: ~/MagangBayu24-ROS2
gusti@gusti-ubuntu:~/MagangBayu24-ROS2$ mkdir tugas3 tugas3/src
```

2. Langkah kedua yaitu create package pada direktori src, disini saya menggunakan python



```
gusti@gusti-ubuntu: ~/MagangBayu24-ROS2/tugas3/src
gusti@gusti-ubuntu:~/MagangBayu24-ROS2$ cd tugas3/src
gusti@gusti-ubuntu:~/MagangBayu24-ROS2/tugas3/src$ ros2 pkg create --build-type ament_python tugas3
going to create a new package
package name: tugas3
destination directory: /home/gusti/MagangBayu24-ROS2/tugas3/src
package format: 3
version: 0.0.0
description: TODO: Package description
maintainer: ['gusti <gustigratia706@gmail.com>']
licenses: ['TODO: License declaration']
build type: ament_python
dependencies: []
creating folder ./tugas3
creating ./tugas3/package.xml
creating source folder
creating folder ./tugas3/tugas3
creating ./tugas3/setup.py
creating ./tugas3/setup.cfg
creating folder ./tugas3/resource
creating ./tugas3/resource/tugas3
creating ./tugas3/tugas3/__init__.py
creating folder ./tugas3/test
creating ./tugas3/test/test_copyright.py
creating ./tugas3/test/test_flake8.py
creating ./tugas3/test/test_pep257.py
gusti@gusti-ubuntu:~/MagangBayu24-ROS2/tugas3/src$
```

3. Selanjutnya kita buat file pada direktori tugas3/src/tugas3/tugas3, bernama publisher.py
4. Menambahkan dependencies rclpy pada package.xml dan menambahkan entry points Bernama 'turtlelaunch' pada setup.py



```
tugas3 > src > tugas3 > package.xml
4  <name>tugas3</name>
5  <version>0.0.0</version>
6  <description>TODO: Package description</description>
7  <maintainer email="gustigratia706@gmail.com">gusti</maintainer>
8  <license>TODO: License declaration</license>
9
10 <test_depend>ament_copyright</test_depend>
11 <test_depend>ament_flake8</test_depend>
12 <test_depend>ament_pep257</test_depend>
13 <test_depend>python3-pytest</test_depend>
14
15 <exec_depend>rclpy</exec_depend>
16
17 <export>
18 | <build_type>ament_python</build_type>
19 </export>
20 </package>
```

```
tugas3 > src > tugas3 > setup.py > ...
9   data_files=[
10       ('share/ament_index/resource_index/packages',
11        ['resource/' + package_name]),
12       ('share/' + package_name, ['package.xml']),
13   ],
14   install_requires=['setuptools'],
15   zip_safe=True,
16   maintainer='gusti',
17   maintainer_email='gustigratia706@gmail.com',
18   description='TODO: Package description',
19   license='TODO: License declaration',
20   tests_require=['pytest'],
21   entry_points={
22       'console_scripts': [
23           'turtlelaunch = tugas3.publisher:main',
24       ],
25   },
26 ]
```

5. Beralih pada file publisher.py

Pertama-tama import semua module/package yang dibutuhkan disini saya import rclpy, Node, dan Twist dalam package geometry_msgs

```
tugas3 > src > tugas3 > publisher.py > ...
1  import rclpy
2  from rclpy.node import Node
3  from geometry_msgs.msg import Twist
4
```

Selanjutnya membuat object node MinimalPublisher, lalu buat publisher pada object constructor dengan parameter(Twist, '/turtle/cmd_vel', 10) dan set timer callback 1 detik.

```
4
5  class MinimalPublisher(Node):
6
7      def __init__(self):
8          super().__init__("minimal_publisher")
9          self.publisher = self.create_publisher(Twist, '/turtle1/cmd_vel', 10)
10         timer_period = 1 # seconds
11         self.timer = self.create_timer(timer_period, self.timer_callback)
12         self.i = 0
13
14     def timer_callback(self):
```

Selanjutnya buat method timer_callback, disinilah program utamanya. Untuk membuat bentuk seperti pada contoh penugasan, terdapat beberapa tahap, pada tahap pertama yaitu membuat segitiga, Ketika membuat segitiga, ada 5 langkah yaitu teleport – rotate – teleport -rotate – teleport, dengan kecepatan/kecepatan sudut tertentu sehingga dapat membentuk segitiga.

```
13
14     def timer_callback(self):
15         if self.i < 6:
16             if self.i % 2 == 0:
17                 msg = Twist()
18                 msg.angular.z = 2.125
19             else:
20                 msg = Twist()
21                 msg.linear.x = 5.0
```

Pada tahap kedua membentuk setengah lingkaran pada setiap sisi segitiga, dengan tiap langkah yaitu rotate – teleport – rotate – teleport – rotate – teleport. Pada tahap ini Ketika teleport (bergerak secara linear) juga mengalami rotasi (msg.angular.z).

```

22         elif self.i >= 6 and self.i < 12:
23             if self.i % 2 == 1:
24                 msg = Twist()
25                 msg.angular.z = 3.16
26                 msg.linear.x = 7.1
27             elif self.i == 6:
28                 msg=Twist()
29                 msg.angular.z = 0.45
30             else:
31                 msg=Twist()
32                 msg.angular.z = -1.3
33
34         self.i += 1
35         self.publisher.publish(msg)
36

```

Untuk melakukan tahap-tahap di atas, saya membuat counter (self.i), di mana counter ini akan melakukan increment setiap selesai melakukan 1 langkah. Pada akhir method tambahkan baris self.publisher.publish(msg)

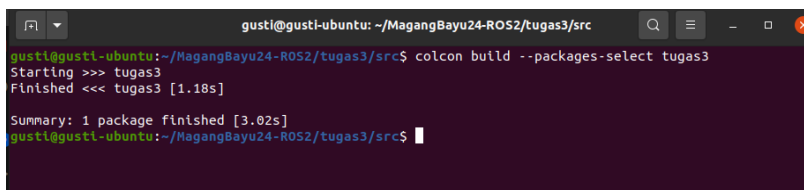
Kemudian buat function main yang akan melakukan spin pada publisher.

```

36
37 def main(args=None):
38     rclpy.init(args=args)
39     minimal_publisher = MinimalPublisher()
40     rclpy.spin(minimal_publisher)
41
42     minimal_publisher.destroy_node()
43     rclpy.shutdown()
44
45 if __name__ == '__main__':
46     main()
47

```

6. Langkah selanjutnya adalah build package



```

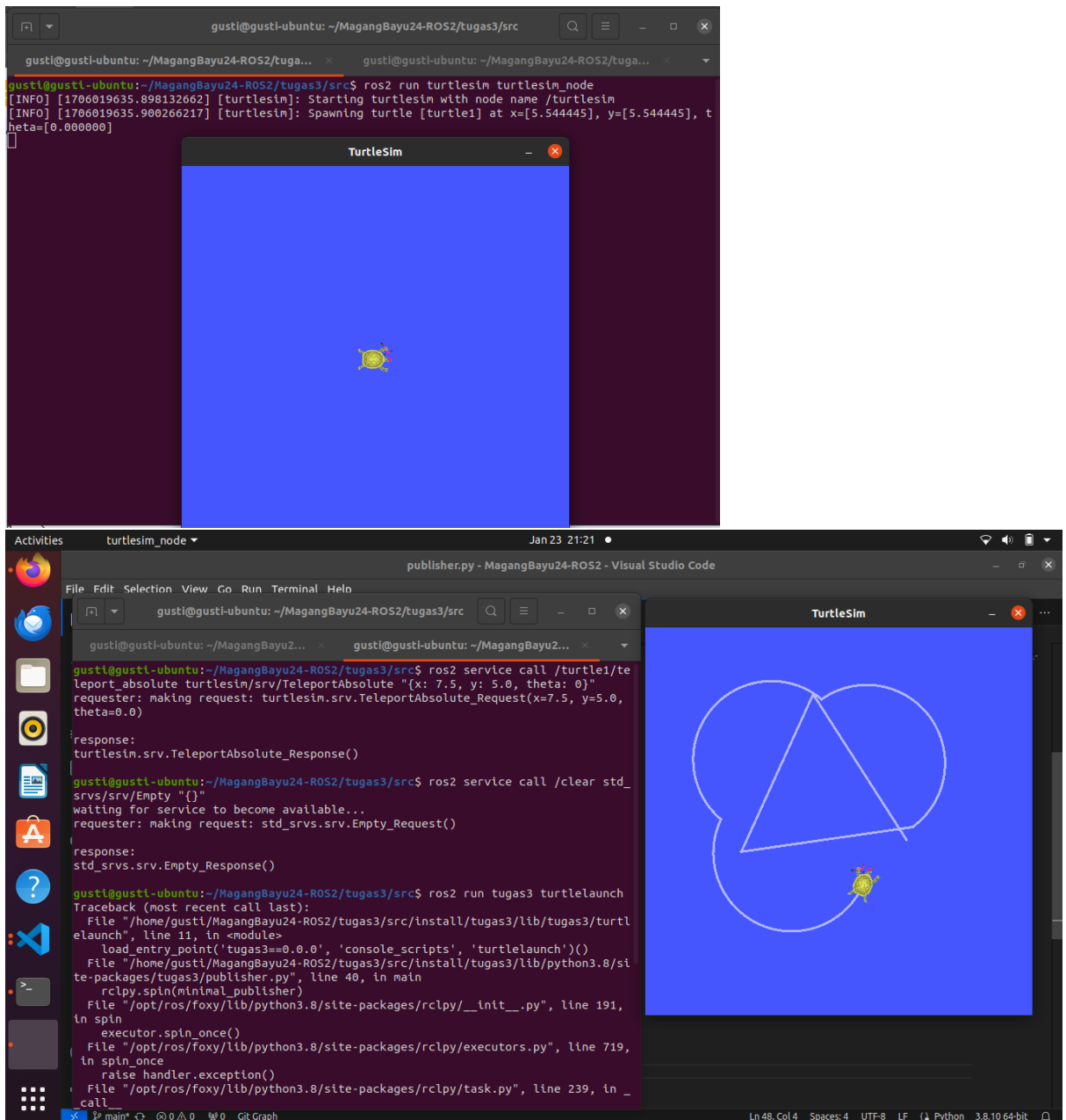
gusti@gusti-ubuntu: ~/MagangBayu24-ROS2/tugas3/src
gusti@gusti-ubuntu:~/MagangBayu24-ROS2/tugas3/src$ colcon build --packages-select tugas3
Starting >>> tugas3
Finished <<< tugas3 [1.18s]

Summary: 1 package finished [3.02s]
gusti@gusti-ubuntu:~/MagangBayu24-ROS2/tugas3/src$

```

- Sebelum run package, buka terminal baru dan run turtlesim_node, lalu di terminal yang lain, sebelum run turtlelaunch gunakan command :
`ros2 service call /turtle1/teleport_absolute turtlesim/srv/TeleportAbsolute "{x: 7.5, y: 5.0, theta: 0}"`

`ros2 service call /clear std_srvs/srv/Empty "{}"`



KENDALA :

Kendala yang saya alami dalam mengerjakan tugas 3 ini adalah saya tidak mengetahui rumus yang harus digunakan dalam membentuk pola yang ditentukan, alhasil saya mengerjakan dengan manual (hardcoded) sehingga pola yang terbentuk jauh dari yang diharapkan tetapi tetap berusaha untuk membuat pola yang semirip mungkin.