

**\*Nama:\* Andiva Kasih Anggoro Putra**

**\*Kelas:\* TK44G7**

**\*NIM:\* 1103204031**

**\*UTS\_Robot\_Autonomy\_Lengan Robot Kinetics Ubuntu\***

\*Youtube Tutorial: [[https://youtu.be/EosEikbZhiM?si=mzig\\_JdrGg8sJ8\\_-](https://youtu.be/EosEikbZhiM?si=mzig_JdrGg8sJ8_-)]  
([https://youtu.be/EosEikbZhiM?si=mzig\\_JdrGg8sJ8\\_-](https://youtu.be/EosEikbZhiM?si=mzig_JdrGg8sJ8_-))

## 1. Overview Nodes\*

Nodes dalam konteks robotika dapat merujuk pada beberapa konsep, yaitu:

### - \*Node dalam Jaringan Robot:\*

Setiap robot atau perangkat dalam sistem dianggap sebagai "node" yang terhubung ke jaringan untuk berkomunikasi dan berkoordinasi satu sama lain.

### - \*Node dalam Struktur Kontrol Robot:\*

Node dapat merujuk pada titik-titik dalam struktur kontrol di mana informasi atau perintah dikirim atau diterima, mengontrol gerakan atau tugas tertentu.

### - \*Node dalam Sistem Sensor atau Aktuator:\*

Node bisa merujuk pada elemen-elemen dalam sistem sensor atau aktuator pada robot, menghasilkan atau mengolah informasi lingkungan atau menggerakkan bagian-bagian robot.

### - \*Node dalam Representasi Model Robot:\*

Dalam pemodelan robot, node bisa merujuk pada representasi titik-titik atau elemen-elemen tertentu dari model robot.

### - \*Node dalam Robotika Bergerak:\*

Dalam robotika bergerak, node dapat merujuk pada posisi atau titik-titik tertentu dalam ruang yang digunakan untuk navigasi atau pemetaan.

## \*2. Overview ROS2\*

ROS2 (Robot Operating System 2) adalah platform perangkat lunak open-source untuk

pengembangan dan pengoperasian robot. Beberapa fitur utama ROS2 meliputi:

- **\*Multi-Platform Support:\***

Mendukung berbagai platform, termasuk Linux, Windows, dan macOS.

- **\*Real-Time Capabilities:\***

Memperkenalkan manajemen waktu yang memungkinkan aplikasi robotika real-time.

- **\*Security:\***

Menyertakan lapisan keamanan untuk autentikasi dan enkripsi komunikasi antar node.

- **\*Pluggable Communication Middleware:\***

Memungkinkan pemilihan middleware komunikasi yang sesuai dengan kebutuhan aplikasi.

- **\*Modular Architecture:\***

Dirancang dengan arsitektur modular, memungkinkan pengguna untuk membangun dan menggunakan berbagai komponen atau "node" yang dapat berkomunikasi satu sama lain.

- **\*Tooling dan Ekosistem Pengembang:\***

Dilengkapi dengan alat pengembangan dan dukungan seperti simulasi dan pemantauan kinerja.

- **\*Documentation dan Komunitas:\***

Memiliki dokumentasi yang kuat dan mendukung komunitas pengembang yang aktif.

### **\*3. Menjalankan Nodes dengan demo\_nodes\_cpp\***

Dalam demo\_nodes\_cpp, contoh pembuatan dua nodes, yaitu talker dan listener:

- Untuk menjalankan talker, ketikkan: `ros2 run demo_nodes_cpp talker`.

- Untuk menjalankan listener, ketikkan: `ros2 run demo_nodes_cpp listener`.

Nodes ini berkomunikasi dan mentransmisikan data "hello world %" antara satu sama lain.

### **\*4. Menjalankan Nodes dengan turtlesim\***

Menggunakan modul turtlesim dengan turtle dan node talker:

- Menjalankan turtlesim node: `ros2 run turtlesim turtlesim_node`.
- Menjalankan turtle\_teleop\_key untuk menggerakkan turtle: `ros2 run turtlesim turtle_teleop_key`.

Data dari talker ditangkap oleh listener dan digunakan untuk menggerakkan turtle dalam simulasi turtlesim.