

Jawaban Soal 1

19624273-Daniel Arrigo Manurung

a.ROS(Robot Operating System)

- Apa itu ROS

Robot Operating System (ROS) adalah *framework open source* terbuka yang membantu peneliti dan pengembang membangun dan menggunakan kembali kode antar aplikasi robotika. ROS juga merupakan komunitas sumber terbuka global yang terdiri dari para insinyur, pengembang, dan penghobi yang berkontribusi dalam menjadikan robot lebih baik, lebih mudah diakses, dan tersedia bagi semua orang.

- Kepentingan ROS

Robot Operating System adalah salah satu *framework* yang paling populer di bidang robotika. ROS penting untuk integrasi berbagai komponen robot seperti sensor, aktuator, dan kamera karena menyediakan kerangka kerja yang memungkinkan komunikasi dan kolaborasi antar komponen secara efisien.ROS juga dilengkapi dengan alat simulasi dan visualisasi yang membantu dalam pengujian dan debugging serta memiliki dukungan komunitas yang kuat.

b.Perbedaan Antara ROS dan ROS2

- Perbedaan Utama ROS dan ROS2

Perbedaan	ROS	ROS2
1.Dukungan Platform	Linux	Lintas Platform
2. Kemampuan Real-Time	Tidak mendukung real-time	Dirancang untuk performa real-time
3. Protokol Komunikasi	TCPROS	Data Distribution Service (DDS)
4.Middleware	Fixed middleware	Mendukung berbagai implementasi middleware
5.ROS Master	Ada	Tidak Ada
6.Multiple Node	Tidak Bisa	Bisa
7.Windows Support	Tidak Ada	Ada

- Keunggulan ROS2

Pengembang cenderung memilih ROS2 untuk proyek baru karena dirancang untuk mendukung aplikasi yang memerlukan pemrosesan real-time, serta menyediakan fitur keamanan bawaan seperti otentikasi dan enkripsi untuk komunikasi yang aman. Selain itu, ROS2 menawarkan dukungan multi-

platform, memungkinkan pengembang untuk bekerja di berbagai sistem operasi seperti Linux, Windows, dan macOS. Penggunaan Data Distribution Service (DDS) dalam ROS2 memberikan pola komunikasi yang lebih fleksibel dan scalable, sementara pengaturan Quality of Service (QoS) memungkinkan penyesuaian keandalan komunikasi sesuai kebutuhan aplikasi. Dengan tetap mempertahankan modularitas yang ada di ROS, ROS2 juga memungkinkan penggunaan kembali kode dan alat, didukung oleh komunitas yang aktif dan banyak sumber daya, menjadikannya pilihan yang lebih menarik untuk pengembangan sistem robotik modern.

c. Simulasi Robotik

- **Kepentingan Simulasi**
Simulasi robotik penting dalam pengembangan robot karena memungkinkan pengujian dan perbaikan desain secara virtual tanpa biaya dan risiko fisik. Keuntungan menggunakan simulasi sebelum membangun robot fisik antara lain menghemat waktu dan biaya, mengidentifikasi kesalahan lebih awal, dan mempercepat proses pengembangan.
- **Contoh**
Contoh adalah dalam pembuatan robot untuk lompat tinggi, simulasi memungkinkan para insinyur untuk menguji berbagai desain dan algoritma kontrol loncatan tanpa harus membuat dan menguji prototipe fisik. Dengan menggunakan simulasi, dampak loncat pada robot dapat dianalisis, sehingga tidak hanya mengurangi biaya material untuk membangun kembali robot yang gagal, tetapi juga mempercepat proses iterasi desain sehingga menjadi optimal.

d. Gazebo

- **Apa itu Gazebo**
Gazebo adalah simulator dinamis 3D dengan kemampuan mensimulasikan populasi robot secara akurat dan efisien di lingkungan dalam dan luar ruangan yang kompleks. Meskipun mirip dengan mesin game, Gazebo menawarkan simulasi fisika dengan tingkat ketelitian yang jauh lebih tinggi, serangkaian sensor, dan antarmuka untuk pengguna dan program.
- **Kegunaan Gazebo**
Gazebo dapat melakukan modeling robot yang mencakup bagian mekanis, aktuator, dan sensor. Gazebo juga dapat digunakan untuk menghitung interaksi fisika seperti gravitasi, berdasarkan model fisik yang ditentukan. Robot juga dapat berinteraksi dengan objek lain di lingkungan.

simulasi, seperti menghindari rintangan, memanipulasi objek, atau bernavigasi.


- Mengintegrasikan ROS 2 dan Gazebo

- 1) Menginstal ROS 2 dan Gazebo



```
sudo apt install ros-foxy-gazebo-ros-pkgs
```

- 2) Source ROS 2




```
source /opt/ros/foxy/setup.bash
```

- 3) Install package **ros_gz** untuk integrasi ROS 2 dengan Gazebo:



```
sudo apt install ros-foxy-ros-gz
```

- 4) Source workspace

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The terminal displays two lines of code: `source /opt/ros/foxy/setup.bash` and `source /usr/share/gazebo/setup.sh`.

```
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo/setup.sh
```

5) Jalankan Gazebo dengan plugin ROS 2:

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The terminal displays a single line of code: `gazebo --verbose -s libgazebo_ros_init.so -s libgazebo_ros_factory.so`.

```
gazebo --verbose -s libgazebo_ros_init.so -s libgazebo_ros_factory.so
```

e. Navigasi Robot

- Konsep-Konsep

Mapping: Proses pembuatan peta lingkungan menggunakan sensor robot (seperti LIDAR) untuk mendeteksi dan mengidentifikasi objek di sekitarnya. Peta ini dapat berupa peta 2D atau 3D yang menunjukkan lokasi dan bentuk objek.

Lokalisasi: Setelah peta dibuat, robot harus dapat menentukan posisinya dalam peta tersebut. Lokalisasi dapat dilakukan dengan membandingkan data sensor saat ini dengan peta yang ada menggunakan teknik seperti filter Kalman atau partikel.

Path Planning: Setelah lokasi diketahui, robot harus merencanakan jalur dari posisi saat ini ke tujuan.

Obstacle Avoidance: Robot perlu menghindari rintangan yang tidak terduga saat bergerak.

Simulasi: Dalam lingkungan simulasi, software seperti Gazebo dapat digunakan untuk menguji algoritma navigasi tanpa risiko kerusakan pada robot fisik, memungkinkan iterasi cepat dan pengembangan.

f.TF(Transform)

- Apa itu TF
TF adalah paket yang memungkinkan pengguna melacak beberapa frame koordinat dari waktu ke waktu. tf menjaga hubungan antara bingkai koordinat dalam struktur pohon yang disangga dalam waktu, dan memungkinkan pengguna mengubah titik, vektor, dll antara dua bingkai koordinat pada titik waktu yang diinginkan.
- Bagaimana TF membantu robot bergerak
Dalam simulasi, TF digunakan untuk memastikan robot bergerak dengan benar dengan mengatur frame koordinat seperti base_link untuk robot, laser untuk sensor LIDAR, dan camera untuk kamera. Ketika robot bergerak, TF memperbarui posisi base_link dan mengubah data dari sensor LIDAR ke frame tersebut, sehingga informasi jarak dapat diinterpretasikan dengan tepat. Dengan cara ini, algoritma perencanaan jalur dapat menggunakan posisi robot dalam base_link untuk merencanakan gerakan selanjutnya, memastikan semua perhitungan dilakukan dalam frame yang konsisten. TF secara terus-menerus memperbarui transformasi antar frame, sehingga robot dapat bergerak akurat dan merespons perubahan lingkungan dengan baik