

Nama : Puteri Andini Rosmadila

NIM : 1103204014

Kelas : TK44G7

## **Technical Report ROS2 Navigation Humble - Chapter 1**

### **Pendahuluan**

Navigasi adalah salah satu kemampuan kunci yang membuat robot dapat berinteraksi secara efektif dengan lingkungannya. Dalam konteks robotika, navigasi tidak hanya berkaitan dengan perpindahan fisik dari satu tempat ke tempat lain, tetapi juga melibatkan sejumlah proses kompleks yang memungkinkan robot untuk bergerak secara otonom dan menghindari rintangan. Robot perlu memiliki pemahaman yang mendalam tentang lingkungan sekitarnya, mengetahui posisinya dengan tepat, merencanakan rute yang optimal, dan kemudian mengontrol pergerakannya sambil menghindari halangan. Semua langkah ini memerlukan sistem navigasi yang handal. Dalam perjalanan menuju navigasi yang lebih cerdas, Robot Operating System (ROS) memainkan peran krusial dengan menyediakan paket-paket navigasi pra-dibangun. ROS mempermudah proses pengembangan robot dengan memberikan solusi siap pakai untuk pemetaan, lokalitasi, perencanaan jalur, dan kendali robot. Dengan pemahaman yang mendalam tentang navigasi robotika, dapat merancang robot yang mampu beroperasi secara mandiri, menghadapi tugas-tugas kompleks, dan berinteraksi dengan lingkungan sekitarnya secara optimal.

Navigasi memiliki peran krusial dalam meningkatkan kemampuan robot. Berikut adalah beberapa alasan mengapa navigasi menjadi unsur penting dalam kemampuan robot :

- **Pindah dari Titik A ke Titik B**

Navigasi memungkinkan robot untuk berpindah dari satu lokasi ke lokasi lain dengan efisien. Ini sangat penting untuk menjalankan tugas-tugas tertentu, seperti pengiriman barang, eksplorasi lingkungan, atau pemindaian area tertentu.

- **Otonomi**

Dengan sistem navigasi yang baik, robot dapat beroperasi secara otonom tanpa bantuan eksternal. Kemampuan untuk menavigasi dengan sendirinya memungkinkan robot untuk melaksanakan tugas-tugas kompleks dan menjalani operasi tanpa keterlibatan manusia secara terus-menerus.

- **Interaksi dengan Lingkungan**

Navigasi memungkinkan robot untuk berinteraksi dengan lingkungannya secara efektif. Robot yang mampu menavigasi dengan baik dapat menghindari rintangan, mengidentifikasi tujuan, dan menyesuaikan pergerakannya sesuai dengan perubahan dalam lingkungan sekitarnya.

- **Efisiensi Operasional**

Robot yang dapat melakukan navigasi dengan efisien dapat mengeksekusi tugas-tugasnya dengan waktu yang lebih singkat dan dengan penggunaan sumber daya yang lebih hemat. Ini dapat meningkatkan efisiensi operasional dan produktivitas robot dalam berbagai konteks.

## **Cara Robot Menavigasi**

- **Pemetaan (Mapping)**

Seperti manusia yang memiliki gambaran mental atau fisik tentang tempat di sekitarnya, robot memerlukan pemetaan untuk mengetahui struktur dan karakteristik lingkungannya.

- **Lokalisasi**

Mirip dengan manusia yang mengetahui di mana mereka berada, robot perlu melakukan lokalisasi untuk menentukan posisinya dalam pemetaan yang telah dibuat.

- **Perencanaan Jalur (Path Planning)**

Analog dengan manusia yang merencanakan rute terbaik untuk mencapai tujuan, robot perlu merencanakan jalur yang optimal dalam pemetaan untuk mencapai lokasi tujuan tanpa bertabrakan.

- **Kendali Robot dan Menghindari Rintangan**

Setelah jalur direncanakan, robot perlu mengontrol pergerakannya untuk mengikuti jalur tersebut.

## Langkah – Langkah Meluncurkan Skenario Navigasi

- ```
user:~$ cd ~/ros2_ws/src
user:~/ros2_ws/src$
```

1. Menjalankan perintah yang digunakan untuk pindah ke direktori src di dalam direktori ros2\_ws (workspace ROS 2). Biasanya, direktori src adalah tempat di mana menempatkan paket-paket ROS 2 yang ingin digunakan.
- ```
user:~/ros2_ws/src$ git clone https://bitbucket.org/theconstructcore/ros2_nav_files.git
fatal: destination path 'ros2_nav_files' already exists and is not an empty directory.
```

2. Menjalankan perintah `git clone https://bitbucket.org/theconstructcore/ros2\_nav\_files.git` yang digunakan untuk mengkloning repositori yang berisi paket nav2\_course dari Bitbucket. Repositori ini mungkin berisi paket-paket yang diperlukan untuk demo atau proyek tertentu.
- ```
user:~/ros2_ws/src$ source /opt/ros/humble/setup.bash
```

3. Menjalankan perintah `source /opt/ros/humble/setup.bash` yang bertugas mengatur lingkungan kerja ROS untuk menggunakan ROS Humble (mungkin versi ROS tertentu). Ini mengatur variabel lingkungan dan konfigurasi lainnya agar sesuai dengan versi ROS yang diinginkan.
- ```
user:~/ros2_ws/src$ cd ~/ros2_ws
user:~/ros2_ws$
```

4. Menjalankan perintah `cd ~/ros2_ws` untuk pindah ke direktori utama workspace ROS

2

```
user:~/ros2_ws$ colcon build
Starting >>> nav2_course
Starting >>> nav2_project
/usr/lib/python3/dist-packages/setuptools/dist.py:723: UserWarning: Usage of dash-separated 'script-dir' will not be supported in future versions. Please use the underscore name 'script_dir' instead
  warnings.warn(
/usr/lib/python3/dist-packages/setuptools/dist.py:723: UserWarning: Usage of dash-separated 'install-scripts' will not be supported in future versions. Please use the underscore name 'install_scripts' instead
  warnings.warn(
Finished <<< nav2_course [1.19s]
--- stderr: nav2_project
/usr/lib/python3/dist-packages/setuptools/dist.py:723: UserWarning: Usage of dash-separated 'script-dir' will not be supported in future versions. Please use the underscore name 'script_dir' instead
  warnings.warn(
/usr/lib/python3/dist-packages/setuptools/command/install.py:34: SetuptoolsDeprecationWarning: setup.py install is deprecated. Use build and pip and other standards-based tools.
  warnings.warn(
---
Finished <<< nav2_project [2.07s]

Summary: 2 packages finished [4.02s]
1 package had stderr output: nav2_project
```

5. Menjalankan perintah `colcon build` untuk membangun semua paket di dalam workspace menggunakan Colcon. Colcon adalah tools yang digunakan untuk membangun paket-paket ROS 2.

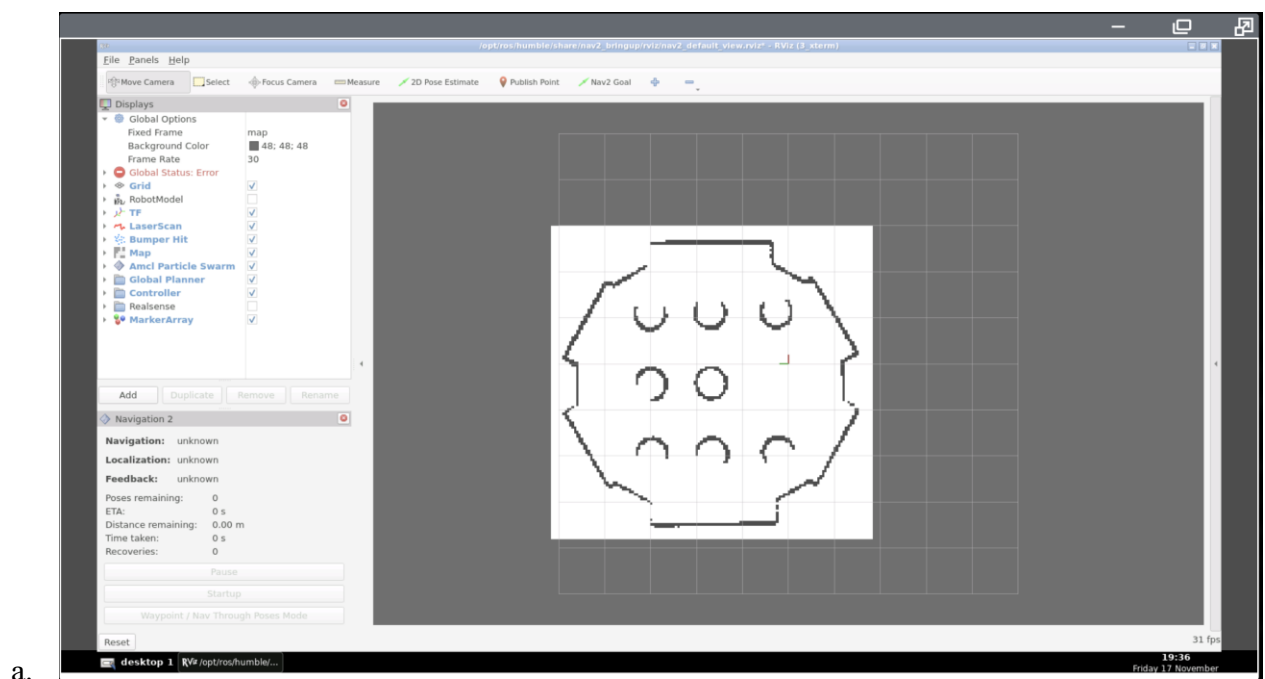
6. `user:~/ros2_ws$ source install/setup.bash`

Menjalankan perintah **source install/setup.bash** setelah selesai membangun, perintah ini mengatur lingkungan kerja untuk menggunakan hasil pembangunan yang baru saja dihasilkan. Ini memastikan bahwa paket-paket yang telah dibangun dapat diakses dengan benar.

7. `user:~/ros2_ws$ ros2 launch nav2_course nav2_demo.launch.py`

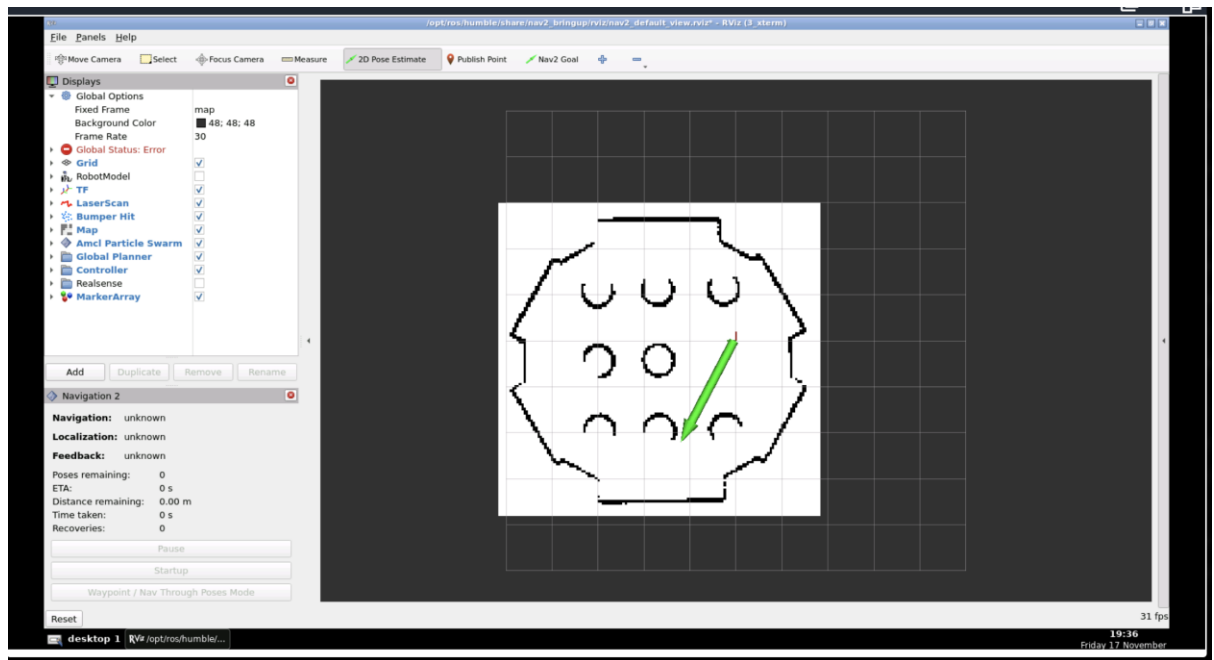
Menjalankan perintah **ros2 launch nav2\_course nav2\_demo.launch.py** untuk meluncurkan skenario navigasi yang disediakan oleh paket `nav2_course`. `nav2_demo.launch.py` adalah file konfigurasi yang digunakan oleh ROS Launch System untuk meluncurkan skenario tersebut.

## Langkah – Langkah Menjalankan Sistem Navigasi Rviz



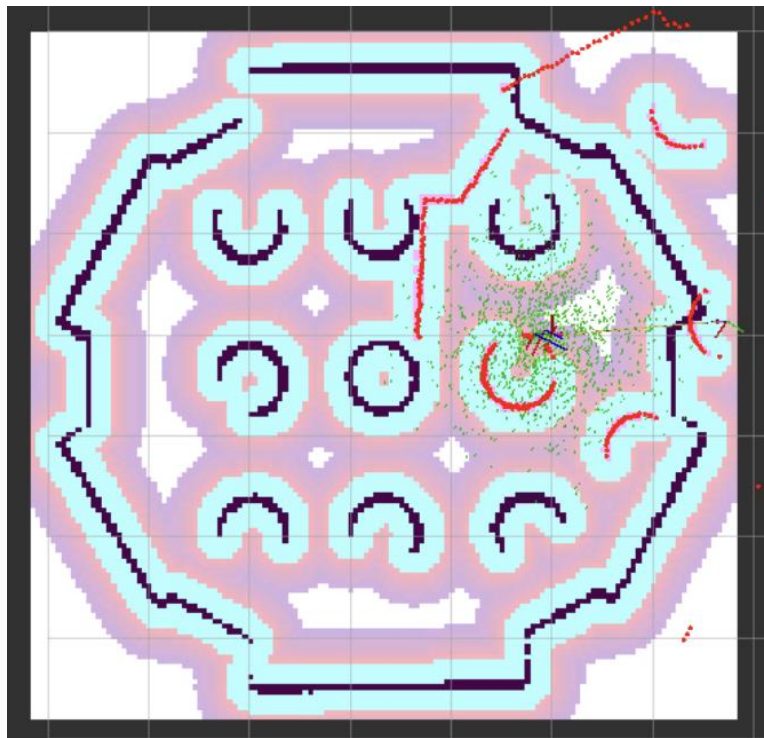
Setelah sistem navigasi Rviz terbuka, memberikan lokasi awal untuk robot.

Caranya dengan klik  pada Rviz.

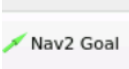


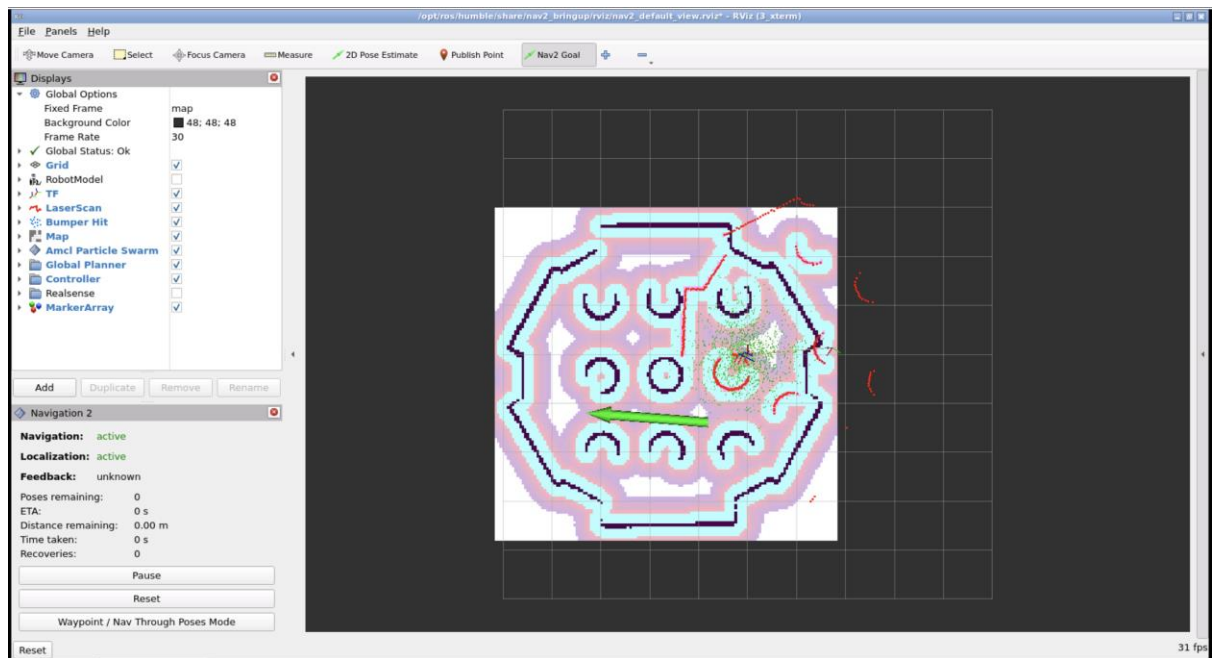
b.

Klik pada peta Rviz di posisi yang menjadi titik awal mulainya robot.



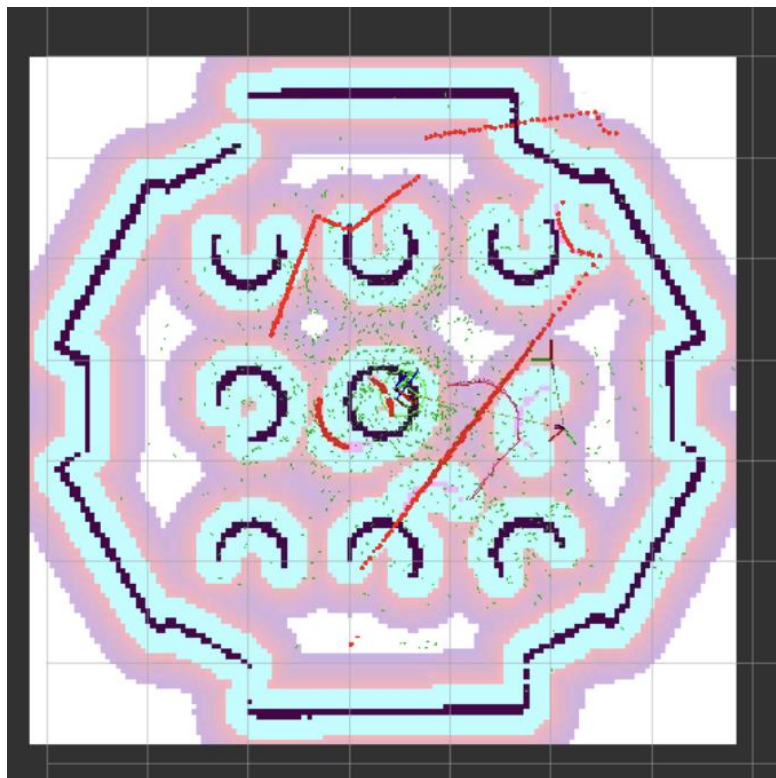
c.

Setelah menentukan titik awal, Klik  untuk menentukan titik tujuan robot agar dapat bergerak mengikuti jalur yang sudah ditentukan tadi.



d.

Klik pada peta Rviz2 diposisi yang akan menjadi titik akhir .



e.

Robot akan bergerak pada sistem navigasi Rviz2