

TECHNICAL REPORT ROBOTICS
ROS2 – CREATE A C++ PACKAGE



Disusun Oleh:

Ratika Dwi Anggraini (NIM: 1103201250)

PROGRAM STUDI S1 TEKNIK KOMPUTER
FAKULTAS TEKNIK ELEKTRO
UNIVERSITAS TELKOM

2023

I. PENDAHULUAN

Robot Operating System 2 (ROS 2) merupakan suatu framework yang powerful dan populer untuk pengembangan robotika. Dengan kemampuannya yang modular dan mendukung berbagai bahasa pemrograman, ROS 2 mempermudah para pengembang dalam membangun sistem robotika yang kompleks. Laporan teknis ini membahas langkah-langkah dalam membuat package menggunakan bahasa pemrograman C++ di dalam ROS 2.

II. LATAR BELAKANG

ROS 2, atau Robot Operating System 2, adalah sebuah framework yang dirancang khusus untuk pengembangan robotika yang modular dan mendukung berbagai bahasa pemrograman. Dibangun sebagai penerus dari ROS yang terkenal, ROS 2 membawa sejumlah fitur baru, termasuk peningkatan dalam keandalan, dukungan untuk komunikasi real-time, dan kemampuan untuk beroperasi di berbagai lingkungan komputasi. Fitur-fitur utama ROS 2 meliputi middleware komunikasi yang dapat dipilih, manajemen package yang lebih baik dengan alat build ament, serta kemampuan untuk mengembangkan solusi robotika yang lebih kompleks. Konsep kunci di dalam ROS 2 adalah "package" yang merupakan unit dasar penyusun sistem. Package adalah cara untuk mengorganisir dan mengelola kode, menyediakan struktur yang terstruktur untuk pengembangan dan pemeliharaan yang efisien.

III. PERSYARATAN

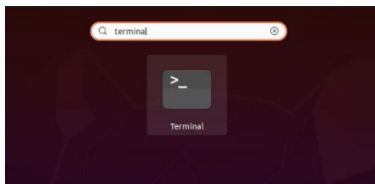
Sebelum memulai pembuatan C++ package di ROS 2, sejumlah persyaratan perlu dipenuhi untuk memastikan pengembangan berjalan lancar. Pertama-tama, pastikan bahwa instalasi ROS 2 telah berhasil dilakukan. ROS 2 mendukung beberapa distribusi, seperti "Foxy," "Galactic," dan sebagainya. Pastikan untuk mengikuti panduan instalasi yang terbaru dari situs resmi ROS 2 untuk distribusi yang sesuai dengan kebutuhan. Selain itu, memastikan bahwa ROS 2 dapat diakses melalui terminal dan variabelnya terkonfigurasi dengan benar. Pastikan juga alat-alat pendukung seperti `colcon` dan `ament` telah terpasang, karena ini diperlukan untuk proses build dan manajemen package di ROS 2. Terakhir, pastikan sistem operasi dan lingkungan pengembangan yang digunakan sesuai dengan rekomendasi ROS 2 untuk meminimalkan potensi konflik atau kendala selama pengembangan.

IV. SET UP ROS2 WORKSPACE

Proses menyiapkan ruang kerja ROS 2 yang baru dimulai dengan pembuatan direktori khusus yang akan berfungsi sebagai workspace. Workspace ini akan menjadi tempat penyimpanan semua package, konfigurasi, dan file-file terkait pengembangan ROS 2. Langkah pertama adalah membuat workspace menggunakan alat 'colcon', yaitu perintah 'colcon build' akan digunakan untuk membangun package di dalam workspace. Selanjutnya, pastikan untuk mengatur variabel lingkungan dengan menjalankan script 'source' yang disediakan oleh ROS 2, sehingga terminal yang digunakan dapat mengenali perintah-perintah ROS 2. Terakhir, verifikasi bahwa instalasi ROS 2 berfungsi dengan menjalankan perintah sederhana seperti 'ros2 doctor', yang memberikan pemeriksaan cepat tentang konfigurasi dan dependensi yang diperlukan.

V. INSTALASI

1. Membuka terminal pada Ubuntu



2. Memasang python3-colcon-common-extensions lalu build ROS2 packages menggunakan colcon build

```
ratika@ratika-VirtualBox: ~/ros2_ws
ratika@ratika-VirtualBox:~$ cd
ratika@ratika-VirtualBox:~$ ls
Desktop  Downloads  Pictures  ros2_ws  Templates
Documents  Music  Public  src  Videos
ratika@ratika-VirtualBox:~$ cd ros2_ws/
ratika@ratika-VirtualBox:~/ros2_ws$ ls
build  install  log  src
ratika@ratika-VirtualBox:~/ros2_ws$ sudo apt install python3-colcon-common-extensions
[sudo] password for ratika:
Reading package lists... Done
Building dependency tree
Reading state information... Done
python3-colcon-common-extensions is already the newest version (0.3.0-1).
python3-colcon-common-extensions set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ratika@ratika-VirtualBox:~/ros2_ws$ colcon build
```

3. Melihat package yang tersedia

```

ratika@ratika-VirtualBox: ~/ros2_ws/src
lates.cmake)
returned error code 1
Call Stack (most recent call first):
/opt/ros/foxy/share/ament_cmake_core/cmake/ament_cmake_coreConfig.cmake:41 (in
clude)
/opt/ros/foxy/share/ament_cmake/cmake/ament_cmake_export_dependencies-extras.c
make:15 (find_package)
/opt/ros/foxy/share/ament_cmake/cmake/ament_cmakeConfig.cmake:41 (include)
CMakeLists.txt:18 (find_package)

---
Failed <<< webots_ros2_msgs [0.95s, exited with code 1]
Aborted <<< webots_ros2_importer [1.26s]

Summary: 0 packages finished [3.61s]
1 package failed: webots_ros2_msgs
1 package aborted: webots_ros2_importer
1 package had stderr output: webots_ros2_msgs
10 packages not processed
ratika@ratika-VirtualBox:~/ros2_ws$ ls
build install log src
ratika@ratika-VirtualBox:~/ros2_ws$ cd src/
ratika@ratika-VirtualBox:~/ros2_ws/src$ clear

```

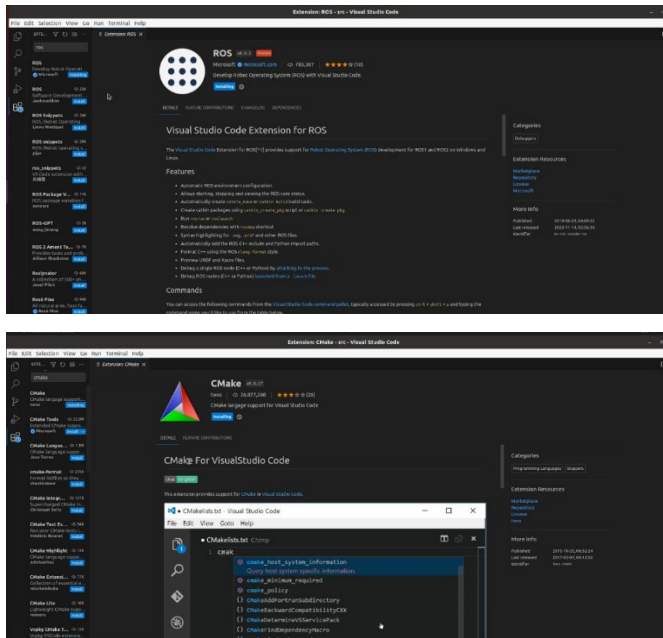
4. Mengunduh C++ package dan Visual Studio Code

```

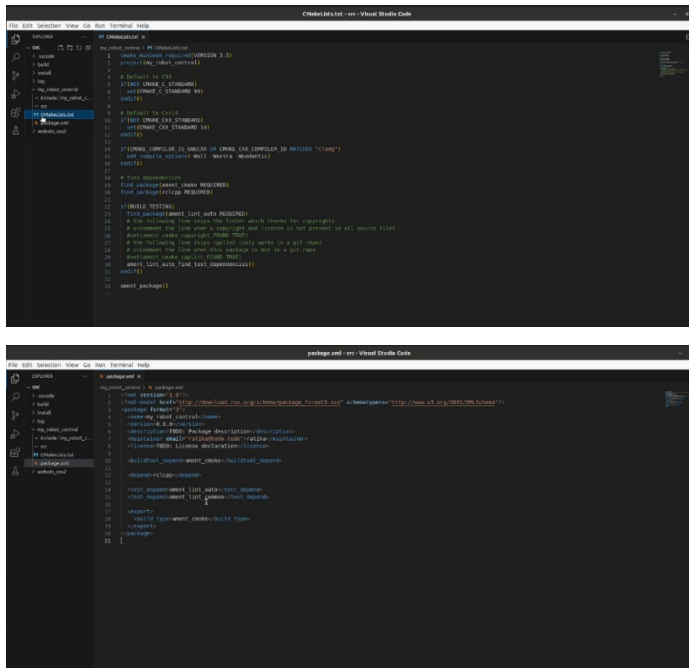
Nov 17 18:33
ratika@ratika-VirtualBox:~/ros2_ws/src
ratika@ratika-VirtualBox:~/ros2_ws/src$ ros2 pkg create my_robot_control --buil
d-type ament_cmake --dependencies rclcpp
going to create a new package
package name: my_robot_control
destination directory: /home/ratika/ros2_ws/src
package format: 3
version: 0.0.0
description: TODO: Package description
maintainer: ['ratika <ratika@todo.todo>']
licenses: ['TODO: License declaration']
build type: ament_cmake
dependencies: ['rclcpp']
creating folder ./my_robot_control
creating ./my_robot_control/package.xml
creating source and include folder
creating folder ./my_robot_control/src
creating folder ./my_robot_control/include/my_robot_control
ratika@ratika-VirtualBox:~/ros2_ws/src$ sudo snap install code --classic
code 1a5daa3a from Visual Studio Code (vscode) installed
ratika@ratika-VirtualBox:~/ros2_ws/src$ code .

```

5. Mengunduh Ekstensi ROS dan CMake



6. Melihat isi package C++



7. Mengecek package yang sudah terunduh

```
ratihagratika-VirtualBox:~$ ls
build install log src
ratihagratika-VirtualBox:~$ cd src
ratihagratika-VirtualBox:~/src$ ls
build install log my_robot_control webots_ros2
ratihagratika-VirtualBox:~/src$ cd my_robot_control/
ratihagratika-VirtualBox:~/src/my_robot_control$ colcon build
Starting >>> my_robot_control
Finished <<< my_robot_control [3.50s]
Summary: 1 package finished [6.33s]
```

VI. KESIMPULAN

Dalam laporan ini, telah terurai langkah-langkah dalam membuat C++ package di ROS 2. Dari pendahuluan yang membahas ROS 2 dan struktur paket hingga pembahasan praktis mengenai pembuatan, pengujian, dan dokumentasi telah dijelaskan secara rinci. Keberhasilan dalam pengembangan robotika berbasis ROS 2 tidak hanya bergantung pada fungsionalitas teknis paket, tetapi juga pada kemampuan untuk menyediakan dokumentasi yang jelas, uji yang solid, dan struktur kode yang terstruktur. Dengan mematuhi praktik-praktik ini, dapat dipastikan bahwa kontribusi tidak hanya dapat digunakan secara internal, tetapi juga dapat diadopsi dan dikembangkan oleh komunitas ROS 2 secara lebih luas, menciptakan ekosistem yang kuat dan inovatif dalam pengembangan robotika.