

## Chapter 3

### Langkah-Langkah Chapter 3: Menjalankan View Demo

Pastikan Struktur Workspace Folder Chapter3 harus berada di direktori src dalam workspace Anda. Jika belum ada, pindahkan folder Chapter3:

```
mv ~/Downloads/repository/build/Chapter3 ~/Downloads/repository/src/
```

Pastikan struktur folder seperti berikut:

```
repository/  
├── src/  
│   ├── Chapter3/  
│   │   ├── mastering_ros_robot_description_pkg/  
│   │   │   ├── launch/  
│   │   │   └── ...  
├── build/  
└── devel/
```

Periksa File view\_demo.launch Pastikan file view\_demo.launch ada di dalam folder launch dari package mastering\_ros\_robot\_description\_pkg:

```
ls ~/Downloads/repository/src/Chapter3/mastering_ros_robot_description_pkg/launch
```

Jika file tidak ditemukan, cari di seluruh folder repository:

```
find ~/Downloads/repository -type f -iname "view_demo.launch"
```

Jika file ditemukan di lokasi lain, pindahkan ke folder launch:

```
mv <lokasi_file_view_demo.launch>
```

```
~/Downloads/repository/src/Chapter3/mastering_ros_robot_description_pkg/launch/
```

**Build Workspace** Setelah memastikan file berada di lokasi yang benar, build ulang workspace Anda:

```
cd ~/Downloads/repository
```

```
catkin_make
```

```
source devel/setup.bash
```

**Jalankan File Launch** Jalankan file `view_demo.launch` menggunakan perintah berikut:

```
roslaunch mastering_ros_robot_description_pkg view_demo.launch
```

**Troubleshooting** Jika file masih tidak dapat dijalankan:

Periksa apakah package dikenali oleh ROS:

```
rospack list | grep mastering_ros_robot_description_pkg
```

Pastikan file `CMakeLists.txt` dan `package.xml` di dalam `mastering_ros_robot_description_pkg` terkonfigurasi dengan benar.

Periksa log error ROS:

```
cat ~/.ros/log/latest/roslaunch-*.log
```

---

## Verifikasi Instalasi

Pastikan ROS telah terinstal dengan menjalankan:

```
roscore
```

Jika instalasi berhasil, Anda akan melihat output yang menunjukkan bahwa roscore berjalan tanpa error.

Gunakan perintah berikut untuk memeriksa paket-paket yang terinstal:

```
rospack list
```

---

### Catatan Penting

Selalu pastikan Anda menjalankan perintah berikut di terminal baru sebelum menggunakan ROS:

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

Jika menggunakan workspace, aktifkan setup workspace Anda:

```
source ~/catkin_ws/devel/setup.bash
```

## Analisis Chapter 3

Dari ketiga gambar di atas, dapat dilihat penggunaan RViz (ROS Visualization) yang merupakan tool visualisasi 3D dalam ekosistem ROS. RViz menampilkan berbagai komponen robot dengan sistem koordinat yang berbeda-beda, ditandai dengan tampilan sumbu x (merah), y (hijau), dan z (biru). Gambar-gambar tersebut menunjukkan visualisasi frame transformasi (tf) yang menghubungkan berbagai bagian robot, seperti base\_link, tilt\_link, pan\_link pada gambar pertama, kemudian struktur lengan robot dengan frame shoulder\_pitch\_link, elbow\_roll\_link, dan gripper pada gambar kedua, serta frame roda (wheel) pada gambar ketiga.

Tampilan antarmuka RViz juga menunjukkan berbagai pengaturan visualisasi yang dapat dikonfigurasi, seperti Fixed Frame yang diatur ke base\_link, Background Color dengan nilai RGB 122,122,122, dan Frame Rate 30 fps. Panel sebelah kiri memperlihatkan Grid dengan berbagai parameter seperti Cell Size, Line Style, dan Reference Frame yang dapat disesuaikan. Bagian bawah interface menampilkan informasi waktu ROS dan Wall Time, serta status sinkronisasi yang diatur "Off". Keseluruhan visualisasi ini sangat penting dalam pengembangan robot karena memungkinkan developer untuk memahami posisi relatif setiap komponen robot, memeriksa transformasi antar frame, dan memvalidasi pergerakan robot secara visual sebelum diimplementasikan pada robot fisik.

