```
mkdir -p ~/clean_workspace/src
cd ~/clean_workspace/src
catkin_init_workspace

cd ~/clean_workspace
catkin_make

source devel/setup.bash

cd ~/clean_workspace/src
catkin_create_pkg prm_package_clean rospy visualization_msgs geometry_msgs

mkdir -p ~/clean_workspace/src/prm_package_clean/config
mkdir -p ~/clean_workspace/src/prm_package_clean/launch
mkdir -p ~/clean_workspace/src/prm_package_clean/scripts

nano ~/clean_workspace/src/prm_package_clean/config/params.yaml

prm_params:
  num_nodes: 20          # Jumlah node acak yang akan dihasilkan
  x_limit: 10            # Batas area sumbu X
  y_limit: 10            # Batas area sumbu Y
  max_distance: 3        # Jarak maksimal antar node untuk menghubungkannya

nano ~/clean_workspace/src/prm_package_clean/launch/prm.launch

<launch>
    <!-- Membaca parameter dari file YAML -->
    <param name="prm_params" file="$(find prm_package_clean)/config/params.yaml" />

    <!-- Menjalankan node prm.py -->
    <node pkg="prm_package_clean" type="prm.py" name="prm_node" output="screen"/>
</launch>

nano ~/clean_workspace/src/prm_package_clean/scripts/prm.py

#!/usr/bin/env python3

import rospy
import random
import math
from visualization_msgs.msg import Marker, MarkerArray
from geometry_msgs.msg import Point

def euclidean_distance(node1, node2):
    return math.sqrt((node1[0] - node2[0]) ** 2 + (node1[1] - node2[1]) ** 2)

def connect_nodes(nodes, max_distance):
```

```
    edges = []
    for i, node1 in enumerate(nodes):
        for j, node2 in enumerate(nodes):
            if i != j and euclidean_distance(node1, node2) <= max_distance:
                edges.append((node1, node2))
    return edges

def generate_random_nodes(num_nodes, x_limit, y_limit):
    nodes = []
    for _ in range(num_nodes):
        x = random.uniform(0, x_limit)
        y = random.uniform(0, y_limit)
        nodes.append((x, y))
    return nodes

def visualize_prm(nodes, edges):
    marker_array = MarkerArray()

    node_marker = Marker()
    node_marker.header.frame_id = "map"
    node_marker.type = Marker.SPHERE_LIST
    node_marker.scale.x = 0.2
    node_marker.scale.y = 0.2
    node_marker.scale.z = 0.2
    node_marker.color.r = 0.0
    node_marker.color.g = 1.0
    node_marker.color.b = 0.0
    node_marker.color.a = 1.0
    node_marker.id = 0
    node_marker.pose.orientation.w = 1.0  # Quaternion identitas

    for node in nodes:
        pt = Point()
        pt.x = node[0]
        pt.y = node[1]
        pt.z = 0
        node_marker.points.append(pt)
    marker_array.markers.append(node_marker)

    edge_marker = Marker()
    edge_marker.header.frame_id = "map"
    edge_marker.type = Marker.LINE_LIST
    edge_marker.scale.x = 0.05
    edge_marker.color.r = 1.0
    edge_marker.color.g = 0.0
    edge_marker.color.b = 0.0
    edge_marker.color.a = 1.0
    edge_marker.id = 1
```

```
    edge_marker.pose.orientation.w = 1.0  # Quaternion identitas

    for edge in edges:
        pt1 = Point()
        pt1.x = edge[0][0]
        pt1.y = edge[0][1]
        pt1.z = 0
        pt2 = Point()
        pt2.x = edge[1][0]
        pt2.y = edge[1][1]
        pt2.z = 0
        edge_marker.points.append(pt1)
        edge_marker.points.append(pt2)

    marker_array.markers.append(edge_marker)
    return marker_array

if __name__ == "__main__":
    rospy.init_node('prm_visualization')
    pub = rospy.Publisher('visualization_marker_array', MarkerArray, queue_size=10)

    num_nodes = rospy.get_param('prm_params/num_nodes', 20)
    x_limit = rospy.get_param('prm_params/x_limit', 10)
    y_limit = rospy.get_param('prm_params/y_limit', 10)
    max_distance = rospy.get_param('prm_params/max_distance', 3)

    nodes = generate_random_nodes(num_nodes, x_limit, y_limit)
    edges = connect_nodes(nodes, max_distance)

    rate = rospy.Rate(1)
    while not rospy.is_shutdown():
        marker_array = visualize_prm(nodes, edges)
        pub.publish(marker_array)
        rate.sleep()

chmod +x ~/clean_workspace/src/prm_package_clean/scripts/prm.py

cd ~/clean_workspace
catkin_make

source devel/setup.bash

roslaunch prm_package_clean prm.launch

rviz
```