

URO TEST

1.

- a. Apa itu ROS (Robot Operating System), dan bagaimana peran utamanya dalam pengembangan robotik modern? (Jelaskan mengapa ROS penting untuk integrasi berbagai komponen robot seperti sensor, aktuator, dan kamera dalam satu sistem yang bekerja harmonis.)

Robot Operating System (ROS) adalah framework perangkat lunak yang didesain khusus untuk mengembangkan aplikasi robotika. ROS menyediakan berbagai fungsi dan alat yang memungkinkan pengembang untuk mengendalikan robot dengan mudah. Dengan ROS, pengguna dapat menghubungkan, mengontrol, dan mengintegrasikan komponen robot dengan lebih mudah. ROS menggunakan struktur semacam graf di mana masing-masing *node*-nya berupa perangkat lunak yang dapat berkomunikasi melalui protokol *publish-subscribe*.

- b. Apa perbedaan utama antara ROS dan ROS2, dan mengapa pengembang cenderung memilih ROS2 untuk proyek baru? (Jelaskan keunggulan ROS2 dibandingkan ROS dalam hal performa, keamanan, dan pemeliharaan jangka panjang.)

ROS 2 menggunakan Data Distribution Service yang dapat menunjang sistem real-time lebih baik daripada ROS 1 yang menggunakan ROS Master-Slave Architecture. ROS 2 juga memungkinkan beberapa node dijalankan secara paralel sehingga lebih dapat memanfaatkan prosesor multi-core.

ROS 1 tidak menyediakan built-in security feature, sedangkan ROS 2 menyediakan built-in security feature yang memanfaatkan plugin keamanan DDS untuk menyediakan enkripsi berbasis kebijakan, otentikasi, dan kontrol akses.

ROS 1 hanya akan mendapatkan dukungan resmi hingga 2025. Artinya, setelah 2025 pengembangan ROS 1 hanya dapat mengandalkan dukungan dari komunitas. Sebaliknya, ROS 2 masih terus dikembangkan.

- c. Mengapa simulasi robotik penting dalam pengembangan robot, dan apa keuntungan menggunakan simulasi sebelum membangun robot fisik? (Berikan contoh kasus di mana simulasi dapat menghemat waktu dan biaya pengembangan robot.)

Simulasi robotik penting dalam pengembangan robot karena dengan simulasi pengembang dapat menguji algoritma dan desain robot pada lingkungan yang aman dan dapat disimplifikasi. Dengan lingkungan yang aman dan disederhanakan, pengembangan dapat dilakukan bertahap dan dengan risiko rendah. Jika pengembangan dilakukan tanpa simulasi, pengembang akan melakukan banyak perakitan dan pembongkaran. Selain itu, simulasi robotik dapat mempermudah pengembang dalam mengetahui letak error dalam robot. Dengan demikian, simulasi dapat menghemat waktu dan biaya pengembangan.

- d. Apa itu Gazebo, dan bagaimana Gazebo digunakan untuk mensimulasikan lingkungan fisik bagi robot? (Jelaskan langkah-langkah dasar mengintegrasikan ROS dengan Gazebo untuk mengontrol robot dalam simulasi.)

Gazebo adalah alat simulasi robotik open-source yang dapat digunakan untuk meniru lingkungan fisik secara realistis untuk mengetahui perilaku robot. Gazebo menyediakan fitur simulasi fisika yang akurat serta penambahan sensor dan objek. Untuk mengintegrasikan ROS dengan gazebo, pertama pengembang perlu menginstal paket gazebo untuk ROS. Selanjutnya, pengembang perlu memperbarui launch file ROS dan Cmake file untuk menampilkan model dalam gazebo. Setelah itu, tambahkan dependensi pada paket gazebo ros. Akhirnya, simulasi dapat dijalankan.

- e. Bagaimana cara kerja navigasi robot di dunia simulasi? (Apa saja konsep dasar seperti mapping dan lokalisasi yang perlu dipahami, dan bagaimana fitur ini dapat diimplementasikan pada robot Anda?)

Navigasi robot dimulai dari pemetaan lingkungan dengan mengolah data dari berbagai sensor. Setelah itu, akan dilakukan lokalisasi yaitu penentuan posisi robot pada peta. Akhirnya, dapat dilakukan perencanaan jalur dengan algoritma misalnya algoritma Dijkstra untuk menentukan rute paling efisien untuk mencapai tujuan.

- f. Apa itu TF (Transform) dalam konteks ROS, dan bagaimana TF membantu robot memahami posisi dan orientasinya dalam ruang tiga dimensi? (Berikan contoh bagaimana TF digunakan untuk memastikan robot bergerak dengan benar dalam simulasi.)

TF adalah suatu paket/package di ROS yang dapat memungkinkan pengguna mengawasi berbagai frame koordinat dari waktu ke waktu. Pengguna dapat mengakses data posisi dan orientasi terkini dan yang lalu. Dengan begitu, pengguna dapat memastikan robot bergerak sesuai rencana.