



**İstanbul Teknik Üniversitesi Robotik Arama Kurtarma Ekibi**

**Görev Raporu**

**Görev Konusu:** Ros Araştırması

**Hazırlayan:** Kıvanç Karaüzüm

**Teslim Tarihi:** 14.11.2023

**Ros Nedir:**

Robotic Operation System (Ros) robotları kontrol etmek için üretilmiş açık kaynak kodlu bir arayüzdür. İçerisindeki geniş kütüphane havuzu sayesinde endüstriyel robotlardan evsel robotlara hatta savunma sanayi alanına kadar geniş bir yelpazede kullanılıyor.

**Ros Robotikte Hangi Amaçla Kullanılır:**

Ros’un anafikri olan “Tekerleği tekrar icat etmeye gerek yok.” fikrine bağlı olarak projelerde kullanımı kaçınılmaz olan basit seviyedeki kodları tekrar yazmak yerine hazırdaki kodları kullanarak zaman geliştirme aşamasında zamandan tasarruf sağlar. Ros aynı zamanda yüksek miktarda bileşenin koordineli çalışmasında kullanır. Ayrıca multi-language bir dil olduğu için farklı üreticilerden alınan farklı arayüzler hatta farklı diller kullanan bileşenlerin birbirleriyle çalışabilmesi sağlar. Yapılan tasarım ve robotların 3D modellerinin de üretebilen Ros yine geliştirme aşamasında hızlı ilerlemek için kullanılır.

**Robotikte Ros Kullanımı Ne Seviyede Olmalı:**

Ros robotik projeleri geliştirmede birçok kolaylık sağlamaktadır. Özellikle projelerin başlangıcında açık kaynak kodlu olması ve geniş kütüphane havuzu sayesinde hızlı bir ilerme sağlar.Yüksek sayıda bileşen bulunan projeler için içerisinde bulunan iletişim modüllerini kullanarak stabil bir iletişim sağlar. Yüksek modülerliği sayesinde takım çalışmalarında bireylerin farklı bölümler üzerinde çalışmasına ve sonra kodlarını birleştirmelerine yardımcı olur. Tüm bunlara rağmen yapacağınız projeye göre Ros kullanımını azaltmanız gerekebilir. Ros’un modülerliği kodları geliştirme kısmında zaman kazandırsa da içerilerindeki fonksiyonlar modülerliği arttırmak için oldukça basit ve genel yazılmışlardır. Bunun yanı sıra Ros büyük ve karmaşık robotik sistemleri çalıştırmak için tasarlanmıştır bu yüzden basit görevler için fazla karmaşıktır. Bunlara ek olarak Ros’un sağladığı bileşenler arası iletişim küçük robotlar için gereksiz ve yavaş olabilir. Tüm bunlar haricinde rosun açık kaynak kodlu olması da bazı sorunlara yol açabilir. Açık kaynak kodlu yazılımlar topluluk tarafından yeteri kadar test edilmediği için hatalar barındıyor olabilir. Ayrıca bu kodlar farklı kaynaklardan alındığı için aralında uyum sorunları meydana gelmesi de olası. Bizim durumumuz için beetle gibi fazla bileşenli robotlarda işleri kolaylaştırırken daha küçük ve hızlı hareket etmesi gereken robotlarda gereksiz ve yavaş olabilir.

**Ros Kullanımı Nasıl Azaltılabilir:**

Projelerde Ros kullanımını azaltmanın birçok yolu var. Ros genel kullanıma yönelik geniş kütüphaneler kullanıyor bunlar yerine daha spesifik kütüphaneler tercih edilebilir. Ros mesaj formatları yerine kendi spesifik kullanımızı yerine yeni mesaj formatları kodlanabilir. Projede kullanılacak bileşenler eğer bir üreticiden alınacaksa uyumluluklarına göre değerlendirilebilirler. Kendi yazdığımız kodlarda mümkün olduğunca platformdan bağımsız diller kullanarak yine ros kullanımını azaltabiliriz.

**Kaynakça:**

[Why don't we use ROS? | Robots for the masses - not the classes - Open Source Pulurobots! (pulurobotics.fi)](https://pulurobotics.fi/blog/pulurobotics-blog-1/post/why-don-t-we-use-ros-7)

[ROS: Robot Operating System Nedir? | Devnot](https://devnot.com/2020/ros-robot-operating-system-nedir/)

[(166) What is ROS? Why it's Important for making Robots! - YouTube](https://www.youtube.com/watch?v=LyC9RAYE96M&ab_channel=CodeandDecode)

[(166) What is ROS, When to use it, and Why ? - Robot Operating System Tutorial - YouTube](https://www.youtube.com/watch?v=8QfI5a7lTKU&ab_channel=LeHaoNhi)