EduGroupe
Accompagner pour réussir

Conception robotique avec ROS

Référence : **PYRT020** Durée : **3 jours (21 heures)** Certification : **Aucune**

Connaissances préalables

- Connaissance générale des systèmes d'informations, d'un langage de développement comme Python ou C++

Profil des stagiaires

- Chefs de projet robotique, développeurs et toute personne souhaitant utiliser le système ROS pour concevoir une application robotique.

Objectifs

- Comprendre l'architecture de ROS
- Savoir choisir et mettre en oeuvre les composants adaptés à un projet robotique
- Les travaux pratiques sont réalisés avec ROS 2

Certification préparée

- Aucune

Méthodes pédagogiques

- 6 à 12 personnes maximum par cours, 1 poste de travail par stagiaire
- Remise d'une documentation pédagogique papier ou numérique pendant le stage
- La formation est constituée d'apports théoriques, d'exercices pratiques et de réflexions

Formateur

- Consultant-Formateur expert Robotique et IA

Méthodes d'évaluation des acquis

- Auto-évaluation des acquis par le stagiaire via un questionnaire
- Attestation des compétences acquises envoyée au stagiaire
- Attestation de fin de stage adressée avec la facture

Contenu du cours

1. Introduction

- Présentation de ROS (Robot Operating System), historique, contraintes, fonctionnalités, licence
- Objectifs des créateurs de ROS
- Point sur les dernières distributions de ROS
- Tour d'horizon des robots compatibles
- Exemples d'utilisation de ROS

2. Architecture

- Principe de fonctionnement de ROS : modularité, légèreté, compatibilité
- Les différents éléments : les nœuds, le master, les topics, les messages, les services
- L'organisation des fichiers, l'installation : les packages, les stacks
- Présentation packages disponibles

3. Mise en oeuvre

- Installation sur Ubuntu
- Configuration de l'environnement
- Présentation du filesystem ROS et de roscd, eosis, rospack
- Premier pas : création d'un package
- Gestion des nœuds : utilisation de roscore, rosnode et rosrun
- Création de topics et de services

4. Développement

- Programmation en python d'outils de publication et réception de messages (topics) et de services et clients simples

5. Exploitation

- Utilisation de fichiers .bag pour stocker des messages et les réutiliser
- Gestion des dépendances entre packages
- Exécution de ROS sur une infrastructure de plusieurs serveurs
- Présentation des librairies principales

6. Extensions

- Présentation des librairies principales

Notre référent handicap se tient à votre disposition au 01.71.19.70.30 ou par mail à referent.handicap@edugroupe.com pour recueillir vos éventuels besoins d'aménagements, afin de vous offrir la meilleure expérience possible.