COURSE DESCRIPTION CARD - SYLLABUS

Course name
Software engineering in robotics [N2AiR1-SSiR>IOwR]

Course

Field of study Automatic Control and Robotics	Year/Semester 1/1
Area of study (specialization) Control and Robotic Systems	Profile of study general academic
Level of study second-cycle	Course offered in Polish
Form of study part-time	Requirements compulsory

Number of hours

Lecture 10	Laboratory classes 20	Other 0
Tutorials 0	Projects/seminars 0	

Number of credit points

3,00

Coordinators	Lecturers
dr inż. Bartłomiej Krysiak bartlomiej.krysiak@put.poznan.pl dr hab. inż. Wojciech Kowalczyk wojciech.kowalczyk@put.poznan.pl	

Prerequisites

A student beginning the course should have basic knowledge of robotic system design, analysis and implementation of automatic control systems in the context of basic control tasks, planning and spatial mapping of the environment. The student should have knowledge of construction, use and properties of controllers, ability to describe dynamics of systems in state space and in input-output form. The student should have the ability to use the C++ programming language at the basic level with the use of object oriented techniques. Moreover, the student should be able to use basic information and communication techniques, know how to obtain information from indicated sources, and should be ready to cooperate in a team.

Course objective

Introduction to the methodology of designing information systems used in robot control and management using selected software platforms. The course will cover issues related to robot control in the field of communication, process management and synchronization, implementation of control systems and object models. In addition, the course will cover issues related to sensory data processing including filtering and fusion issues. Further on, issues of visualization of the environment and its objects will be presented to verify the correctness of software implementation. The student will gain the ability to consciously and constructively use the learned design methods for various tasks in robotics.

Course-related learning outcomes

Knowledge
1. Basic knowledge of the selected architecture of the development environment supporting the implementation of robot software. An overview of the basic libraries and tools that enable software development for robotic applications. A general scheme for managing robotic application software within selected libraries and frameworks that manage the operation of robotic systems. [K2_W10]
2 Knowledge of basic data acquisition and visualization techniques in software used in robotic applications. Knowledge of basic techniques for processing this data to infer correctness of robotic systems. [K2_W10]
3. Knowledge of software management techniques in the context of software versioning, archiving, and collaborative implementation of information systems in robotics. [K2_W3], [K2_W12]
4. Basic issues of communication between data processing units in the context of cooperation of multiple robotic systems. [K2_W3]
Skills
1. The ability to use the selected programming platform in robotics applications. [K2_U2], [K2_U22]
2. Is able to select appropriate software libraries application to considered tasks in robot control. [K2_U1]
3. Is able to use software module management architecture within selected software frameworks used in robot software. [K2_U12]
4. Is able to implement simple programming tasks for selected processes implemented in robot control. [K2_U9], [K2_U24], [K2_U25]
5. Is able to manage software in the context of archiving and versioning. [K2_U9],[K2_U24].
Social competences
1. The ability to work in a team and to be aware of the non-technical effects of design decisions in the field of automatic control systems. [K2_K2],[K2_K3]
2. Awareness of the necessity of professional approach to technical issues and critical evaluation of design choices. [K2_K5],[K2_K1].

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
A) In the scope of lectures, verification of the assumed learning outcomes is realized through assessment of students' knowledge during the final written assessment in a written form.
B) In the case of laboratory classes, the verification of the established learning outcomes is realized through the current assessment of the student's progress in learning, as well as the final evaluation of the quality of the designed software, evaluation of the written report on the implementation of the tasks, as well as evaluation of the answers to the questions related to the task.

Programme content

The course syllabus covers the following topics: architecture of the software framework including breakdown of programming modules, data exchange ports, communication messages and initialization parameters; implementation management from the commentd line of the selected development environment; development of software packages using ready-to-use design formats and scripts that run software packages; use of selected software libraries for software development; simulation and visualization of processes in robot control; use of computational libraries for geometric transformations in tasks of computing position, velocity and acceleration of robots; integration of selected software modules to implement complex tasks in robotics.

Course topics

none

Teaching methods

A) Lectures: multimedia presentation (slides) additionally illustrated by examples given and analyzed on the blackboard.
B) Laboratory exercises: performance of programming tasks in the topics given by the instructor.

Bibliography

Basic
[1] Control system design, G. C. Goodwin, S. F. Graebe, M. E. Salgado, Prentice Hall, 2001
[2] Programowanie. Teoria i praktyka z wykorzystaniem C++, Bjarne Stroustrup, Helion, 2010
[3] Programming Robots with ROS, William D. Smart, O'Reilly Media, Inc, USA, 2015.
Additional
[4] Introduction to Ubuntu Linux, Vimal Kumar v., Vimal Kumar V., 2020

Breakdown of average student's workload

	Hours	ECTS
Total workload	75	3,00
Classes requiring direct contact with the teacher	30	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	45	1,00