Robotic Programming Environments (winter 2023/2024)

MSc Łukasz Janiec

Wrocław University of Science and Technology
Faculty of Electronics, Photonics and Microsystems
Department of Cybernetics and Robotics

October 4, 2023

Wrocław University of Science and Technology

HR EXCELLENCE IN RESEARCH

unite!

University Network for Innovation, Technology and Engineering

Outline of the course plan

Information

MSc Łukasz Janiec

- 310/C-3
- e-mail: `lukasz.janiec@pwr.edu.pl`
- `https://kcir.pwr.edu.pl/~ljaniec/`

Consultation hours, winter 2023/2024

- Thursday 10:00-11:00

Additional consultation hours are arranged on an individual basis.
The need for a consultation should be notified in advance by e-mail with **a full description of the problems** encountered, **a list of questions** and **attempts made to resolve them independently**.

`https://dontasktoask.com/`

Special needs

If you have special needs due to your health, disability, or other reasons that affect how you participate in class or access materials, please let me know in a private chat or email. I'll make sure everyone has an equal opportunity to learn, without lowering standards.

Important

Accommodating special needs doesn't mean making things easier. It's more about ensuring fair educational opportunities for everyone.

Course credit rules

Firstly, a few comments:

- First iteration of the course after Ph.D. Janiak moved to the industry.
- The course material needs to be updated by me and the lecturer, it will happen over the semester.
- For interested - old version of the course is here (**link**).
- More material will be added later to the ePortal course.
- Access to the free Datacamp Classroom will be provided to the students (important for learning Python used later in the course).
- Presented own ROS-based (or similar) robotic project can get you extra credit (e.g. points for 5.5 if sufficiently advanced). Similar with any *new* open-source contributions.
- Up to two properly excused absences from classes are allowed.

Course credit rules

Requirements

- student attendance (!)
- preparation work before the laboratory according to the course materials (external resources, lecture examples, instructions etc.), put it on our department **GitLab** before the class
- evaluation of work **at the end of the laboratory**

Evaluation Score	Percentage
3.0	50%
3.5	60%
4.0	70%
4.5	80%
5.0	90%

Total percentage is calculated as a sum of points from every exercise (usually one per 1–2 classes, from 5 to 10 pts each).

Health and safety in the laboratory

- Use common sense.
- Communicate any hardware or software errors to me or technicians.
- Inform me if you don't feel well.
- Consuming food and drinks near computer equipment is prohibited. Spills can cause damage and create electrical hazards.
- Try to work on the university PC, don't bring your own laptop. Use your LDAP account and our GitLab.
- Unauthorized modifications to electrical equipment are strictly prohibited. Don't pull the PC plugs from the sockets!

Please note that a local **student** account with a **student** password (for emergency login for students without a personal LDAP account) is automatically cleansed of all changes made when logout.

(Gather signatures from students)

What is this course about?

We will learn about the fundamentals needed in the robotic projects.

Some of the key abilities

- component-based software engineering - built from loosely-coupled & reusable modular components that communicate via interfaces
- communication between modules - like MQTT protocol or ZeroMQ, asynchronous messaging library(*"The Bulk of Software Engineering is Just Plumbing"*), with good coding practices (e.g. `pydantic`, Google protocol buffers)
- ROS 2 proficiency - concepts, architecture, popular packages, simulations
- OROCOS*, ROS 1 etc. - some legacy frameworks for (real-time*) control of the robot

Questions?

Feel free to ask questions.