

Developing a platform for the **KUKA iDo social robot**

Geert Mol
Creative Technology
Bachelor Thesis

09/07/2023
University of Twente
Supervisor: Edwin Dertien
Critical Observer: Ali Sadeghi

Table of Contents

1. Introduction	4
1.1 KUKA iDo	4
1.2 Research questions	4
1.3 Outline	5
2. Background	7
2.1 Domain	7
2.2 Hardware Platforms	8
2.3 Software	8
2.3.1 Accessibility	8
2.3.2 Robot Operating System	10
2.4 Relevance of Findings	11
3. Methods and Techniques	12
3.1 Design method	12
3.2 Ideation	13
3.3 Specification	13
3.4 Realisation	13
3.5 Evaluation	13
4. Ideation	15
4.1 Stakeholder Analysis	15
4.1.1 Bob Schradenberg	15
4.1.2 Edwin Dertien	16
4.2 Preliminary Requirements	17
4.2.1 Must Have	17
4.2.2 Should Have	17
4.2.3 Could Have	18
4.2.4 Will Not Have	18
4.3 Developed Concepts	19
4.3.1 Multi-layer Solution	19
4.3.2 Motor Driver Board	20
4.3.3 Onboard Computer	21
4.3.4 Basic Controller	21
4.3.5 Battery System	21
5. Specification	22
5.1 Final Requirements	22
5.2 Final Concept	22
5.2.1 Hardware	22

5.2.2 Software	23
5.2.3 Documentation	24
6. Realisation	25
6.1 Hardware	25
6.1.1 Main PCB	25
6.1.2 Radio Remote	27
6.2 Software	27
6.2.1 Arduino	27
6.2.2 ROS 2	28
6.3 Documentation	31
7. Evaluation and Future Work	33
7.1 Hardware	33
7.1.1 Arduino	33
7.1.2 Ubuntu	33
7.1.3 Battery	34
7.2 Software	34
7.3 Documentation	35
7.3.1 Participant with Programming Experience	35
7.3.2 Participant without Programming Experience	35
7.3.3 General Improvements for Wiki	36
8. Conclusion	37
Appendix A - Transcript Interview Bob Schradenberg	38
Appendix B - Transcript Interview Edwin Dertien	51
References	67

List of Figures

Figure 1: KUKA iDo shown to Angela Merkel at Hannover Messe 2019.	7
Figure 2: Savioke Relay service delivery robot. [1]	8
Figure 3: Difference between normaling code and block code.	9
Figure 4: State chart switch case example.	10
Figure 5: The Creative Technology Design Process [1]	14
Figure 6: Multi-layer platform approach with time icons.	20
Figure 7: Basic Layout of the KUKA iDo base components.	21
Figure 8: KUKA base connections to radio and serial computer.	24
Figure 9: Signals of a quadrature encoder.	27
Figure 10: Arduino Pro Micro pin layout.	27
Figure 11: Two way data serial protocol.	31
Figure 12: ROS Network containing PS3 controller and serial communication.	31

1. Introduction

1.1 KUKA iDo

The University has acquired four KUKA iDo 'social' robots that were showcased at the Hannover Messe in 2019. The robots were designed to entertain guests by bringing coffee, fresh air, or taking selfies. The goal of this project is to revive the robots and find a useful purpose for them. The main chosen purpose is to create a platform that makes it possible to easily control the robot, program behaviour, and implement new sensors and actuators. This enables educators, researchers, and students to use these robots for new educational and demonstrational purposes.

Creating these types of robots costs a lot of time and is quite complicated. During robotics projects, students often create robots similar to previous years. All these robots get stored somewhere and are often never used again. This is not very sustainable, so by creating a platform with a professional robot connected, the students will be able to learn about complicated robotics in an environmentally friendly way. The task is to make the robots easily programmable by students during an eight-week project, without requiring much prior knowledge about robotics or programming. This means the robot should have an easy programming interface to implement complex behaviours. Including clear documentation.

KUKA builds complex robot technology primarily focusing on the automotive industry. Especially with the iDo project, the focus has not been on accessibility, modularity, and re-usability of the work. These aspects are important when using a similar system in an educational context. During this project, a new hardware layer will be built that will allow easier access and integration with system interfaces like ROS and Arduino.

1.2 Research questions

In order to create a platform for the KUKA iDo social robots, a set of research questions have been created.

Main-RQ: How to design a platform for the KUKA iDo social robot that makes it possible for a student to use it for educational and demonstrational purposes in an easy manner?

To answer this question, research has to be done on state-of-the-art platforms currently available. This leads to the following sub-question:

Sub-RQ1: Which strategies are currently used to make high-level robot behaviour prototyping accessible?

To make sure students understand how to use the platform for many years to come, documentation and tutorials have to be created. These guides should be clear without requiring hours of reading. This leads to the final sub-question:

Sub-RQ2: What documentation and tutorial types are the best for students to quickly understand and utilise the robot platform?

Combining the answers to these questions should lead to a clear strategy for designing a platform for the KUKA iDo robots.

1.3 Outline

This report will follow the following steps in order to answer the research questions and get to a well-designed end product.

- Chapter 2: Background Research
This chapter will include a description of performed background research including a reflection on the relevance of the findings.
- Chapter 3: Methods and Techniques
This chapter will include a description and argumentation of the method and techniques.
- Chapter 4: Ideation
This chapter will include a stakeholder analysis. From these stakeholders, different requirements will be determined and combined with determined requirements from Chapter 2. Furthermore, concepts will be presented from which a final concept will be chosen.
- Chapter 5: Specification
This chapter will contain the final requirements and the scoping of the project. From these final requirements, a final concept will be presented which can be carried out during this thesis.
- Chapter 6: Realisation
This chapter will describe the implementation process of the chosen specifications. This includes the hardware, software, and documentation.
- Chapter 7: Evaluation and Future Work
This chapter presents the evaluation of the hardware, software, and documentation. Including ideas for future improvements and research.

➤ Chapter 8: Conclusion

In this chapter an overview of the research is given and the relevance of the work is outlined.

2. Background

In Figure 1 the KUKA iDo social robot can be seen at the Hannover Messe in 2019. The visible head of the KUKA is one of many version KUKA created. This particular one is able to take selfies using a DSLR camera built into the head. Furthermore, there are versions containing a coffee machine, air fresheners, a Nintendo Switch, and a light orb.



Figure 1: KUKA iDo shown to Angela Merkel at Hannover Messe 2019.

2.1 Domain

As stated above, the KUKA iDo robot was designed as a social robot, which is not KUKA's usual product focus. Their normal products are meant for production and quality control, so the iDo was an experiment to test if they could break into the social robot market. For the base of the KUKA, they used an existing product from the company Savioke. They make a widely used robot called the Relay, which is a transport robot used in healthcare, food service and real estate environments. [1] (Figure 2) This shows that the technology in social, service, and distribution robots is similar and can be developed for multiple domains. The focus of this research is to use the KUKA iDo as a social or service robot for education. Thus the most important factors are accessibility and a short design cycle to enable quick prototyping during the limited time educators, researchers, and students have.



Figure 2: Savioke Relay service delivery robot. [1]

2.2 Hardware Platforms

The University of Twente has used multiple robots for research purposes before. Platforms like the NAO [2], Pepper [3] and MiRo-E [4] have been used for research in the social interaction domain. The NAO and the Pepper are both designed by a company called SoftBank Robotics (previously Aldebaran) and both come with a software suite called Choregraphe which allows you to create animations, behaviours, and dialogues using a flowchart-type programming interface. The suite allows you to test your code on a simulated robot, or on the real robot. Furthermore, the software is expandable with your own Python code for even more customization options. The Miro-E uses an environment called MiroCode, which uses coding blocks instead of a flowchart, but also has a digital simulator. These platforms can be used as inspiration when choosing or creating our own low-code environment.

2.3 Software

2.3.1 Accessibility

In the industrial robotic industry, accessibility or low-code environments are often not the focus, but research has been conducted to enable the accessible programming of industrial robots. As mentioned above, statechart editors and code blocks are an example, other used techniques in the robotic industry are learning from demonstration and augmented reality.

Codeblocks

Codeblocks have been a reliable technique for teaching students the basic concepts of computer programming and logical thinking. As seen in Figure 3, it is based on removing the complicated syntax necessary to use a conventional programming language by dividing logic into code blocks. These blocks can be clicked and dragged into an editor to create complex logic. This technique prevents syntax errors and makes it possible for the programmer to scroll through a library of available code blocks instead of needing knowledge about all possibilities in a language. [5]

Normally this technique is used to teach novice students below the age of 18. But as Weintrop et al. [6] show, there is potential for using code blocks to program industrial robots. In this study, 67 adults divided into two groups were asked to program an industrial robotic arm using code blocks and conventional programming. The results showed that the people who used code blocks were faster without a loss of accuracy, and with improved usability and overall satisfaction.

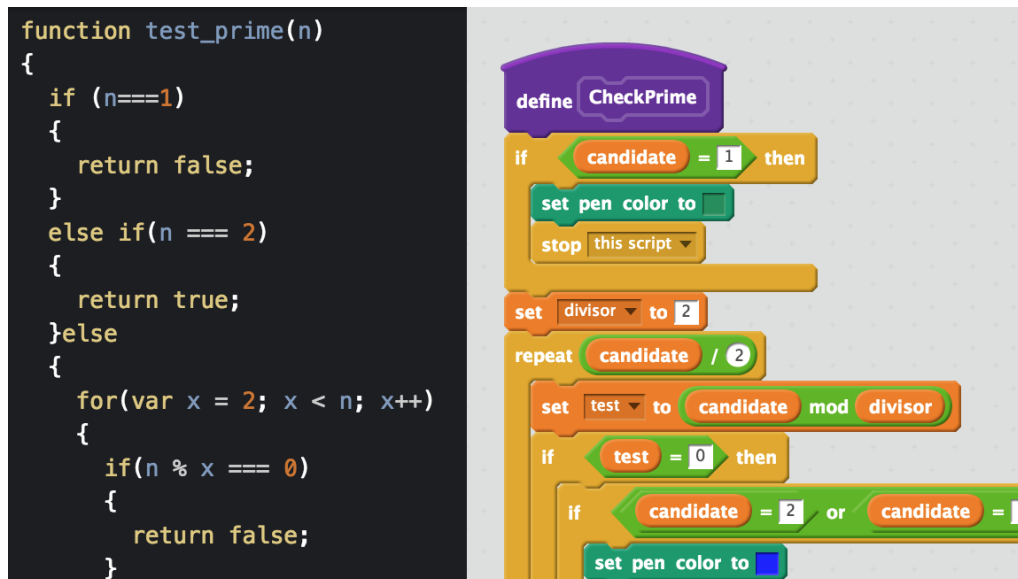


Figure 3: Difference between normaling code and block code.

Statechard Editor

A statechart diagram is a widely used diagram to model the dynamic nature of a system. This system tries to define logic in a flow like diagram. (Figure 4) It defines different states of a system during its operation. These states are stored and can be changed by events. For the Robot Soccer World Cup in 2012 a Computer-Aided Software Engineering (CASE) tool was developed to aid developers while creating complex robot behaviour. This is useful since the teams have a limited amount of time to develop their robots, thus making a faster way to develop the robot behaviour extremely valuable. In this tool, the developer could use a graphical statechart editing

environment utilizing a variety of robot functionalities (vision, localization, motion skills, communication) and connect them using lines and event definitions. The source code would be created automatically which allows people without programming skills to create complex behaviour patterns [7, p232].

The Technical University of Crete tested its CASE tool by running several lab sessions where students were asked to develop an attacker behaviour protocol. The users were then asked to anonymously rate the tool, which resulted in an overall positive result, even though most students were unfamiliar with Agent-Oriented Software Engineering [7, p233].

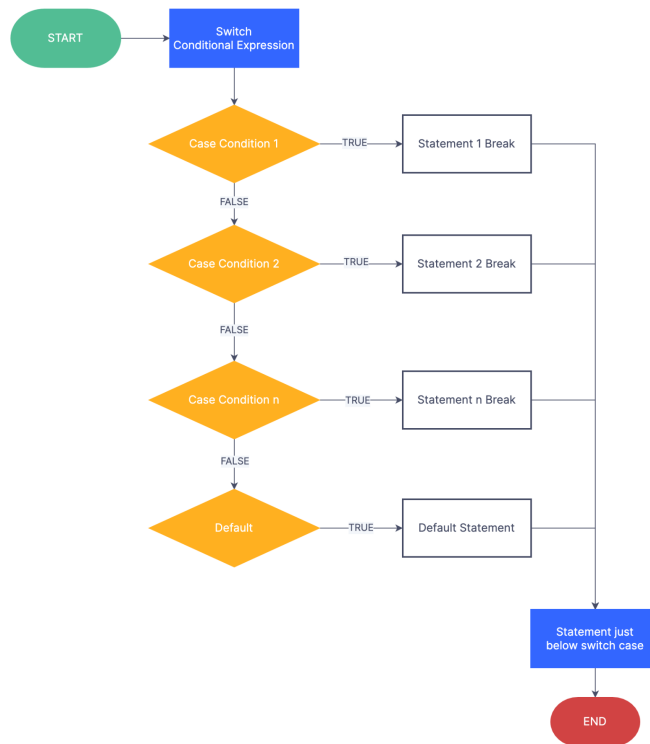


Figure 4: State chart switch case example.

Learning from demonstration

Many robotic applications rely on the mapping between the world-state and actions the robot can take. These mappings are called policies and allow a robot to select an action based on the current world state. Creating these mappings is often challenging and time-consuming so a different approach to policy learning is available, Learning from Demonstration (LfD) [8]. LfD often does not require expert domain knowledge which enables non-experts to develop behaviour policies. Furthermore, demonstration is an intuitive medium we already use to teach other humans.

Learning from Demonstration is becoming more common in assembly line processes, where robots follow a specific predefined path to assembly for example an IKEA chair [9]. If an assembly process changes, the robot needs to be fully reprogrammed and tested. LfD allows a teacher to show the robot new processes by moving the robot physically or in a simulation. From these examples, a neural network is trained which can be implemented to predict the action necessary when a similar object is presented the next time.

Suay et al. [10] Conducted a test using three different LfD algorithms (Behavior Networks, Interactive Reinforcement Learning, and Confidence-Based Autonomy) where non-experts had to create a basic moving pattern using the different algorithms. Users reacted positively to all algorithms and were able to create the movement in a relatively small amount of time. One downside of the algorithms was that the robot policies were not transparent thus making it hard to repair wrong decision policies taken by the robot. An interesting line of research is creating a transparent LfD algorithm that shows the user the robot policies in an intuitive manner.

Augmented Reality

This concept builds upon the Learning from Demonstration topic by adding a programming interface using Augmented Reality. This is useful in the field of human-robot collaboration, this enables humans and robots to work together on a single task. Blankemeyer et al. [11] state that this is already being used in the industry using for example the Microsoft HoloLens. This product combines AR goggles and a camera system in a single device which creates the ability to track physical objects and overlay digital images over them. During an assembly process, the operator can use hand gestures to assemble a product, the HoloLens will turn these gestures into tasks for the robot to perform.

The main problem with this approach is accuracy. Since this method relies on object and hand tracking using cameras a relatively high degree of inaccuracy is measured [11]. Assembly robots require a very high degree of accuracy to increase repeatability and decrease the chance of production failures. Industry-standard assembly robots are often accurate within 0.2 mm [12], while this approach showed to only have a reliable accuracy of 1 mm - 2 mm and a precision of 3 mm - 5 mm. Interesting research for the future would be to develop a high-accuracy tracking system that enables the operators to place objects accurately within current industry standards.

2.3.2 Robot Operating System

The state-of-the-art software for robotics is a platform called ROS. ROS (Robot Operating System) is an open-source software development kit for robotic applications. It is the norm for teaching robotics and the basis for most robotics research. [14] ROS also has libraries to interface with the various sensors of the KUKA. The release of the brand new ROS 2 platform, which is also open source, shows that this platform will stay the industry standard for a long time. For this reason, ROS or ROS 2 is the obvious choice for integrating all the components of the KUKA iDo.

Another benefit of using ROS is its official documentation and tutorials. The tutorials supplied by ROS offer a great way of learning ROS step by step and getting a thorough understanding of the different features it offers.

2.4 Relevance of Findings

The research was conducted with the impression that a fully usable user-friendly programming environment would be developed during this thesis. But as the realisation phase will show, implementing the new hard- and software for the basic control of the KUKA robot was already extremely time consuming. The research shown on this topic is still useful when the KUKA platform will be further developed in the future. Furthermore, the information shown about currently existing platforms, their software and how they organise their documentation applies well to the current project and served as a guide when creating similar systems.

3. Methods and Techniques

3.1 Design method

For this project, the Creative Technology Design method by Mader and Eggink [13] has been chosen (see Figure 5). This process is split into four phases: ideation, specification, realization, and evaluation. The graph shows the process and how the phases relate to one another. The iterative nature of this method makes it possible to always revert back to an earlier phase. This is useful since the robot platform will be improved based on feedback from stakeholders. At the start of the project, a full trial loop of this method has been conducted by using the KUKA robots during a demonstration of the Robotics Centre launch. This resulted in the quick identification of problems and improvements.

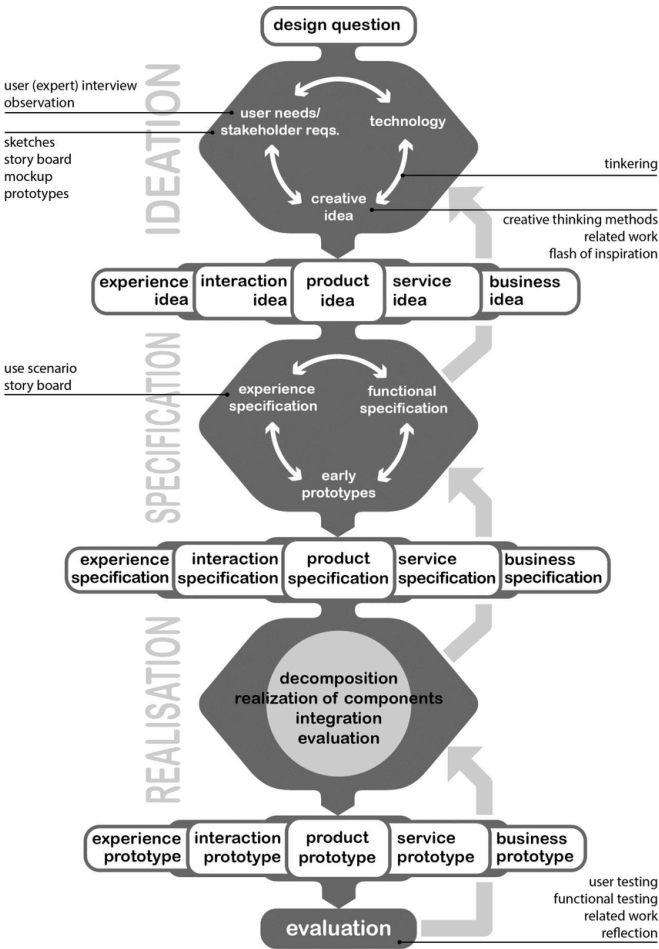


Figure 5: The Creative Technology Design Process [1]

3.2 Ideation

The first phase in the Creative Technology Design process is the ideation phase. The goal is to generate several initial concepts. The first step of this process is doing a literature review of existing techniques of robotic platforms that make behaviour programming accessible through user-friendly interfaces. Literature research will further be used throughout the project wherever it is necessary, this can consist of researching for example robot architecture or robot evaluation techniques. The next step of the ideation phase is holding stakeholder interviews. This will be done with lecturers of the courses Social Robot Design and Human-Robot Communication and a researcher of the Soft Robotics Lab. During these interviews information will be gathered related to the functionalities of the robot, the different 'layers' the platform will consist of, the user interface that will be used, the eventual purposes of the robot platform, and any other relevant topics that might come up.

3.3 Specification

Using the information gathered in the ideation phase a list of requirements will be created using the MoSCoW framework, this framework separates the requirements into four categories, consisting of 'must have', 'should have', 'could have', and 'will not have'. This way the project can be finished in a good manner without all the requirements being met. The second stage of the specification phase is determining the scope of the project. During the ideation phase, many ideas and topics will be created, but not everything will be feasible to realise within one Bachelor Thesis.

3.4 Realisation

Using the acquired specifications, the prototyping phase can begin. During this phase, the hardware will be tested and modified and software prototypes will be created, which will show what parts of the requirements are feasible and what parts need rethinking, resulting in a circular design process. To test the education benefit of the robot platform a basic layout for the documentation will be created. This will be modified later in the project with information gathered in the evaluation phase.

3.5 Evaluation

During this phase, the prototype will be tested on two fronts, the usability of the platform in its final form using driving tests and the documentation provided to modify the platform. The modifiability will be tested by giving students a guide on how to set up the platform and giving them the task to create a basic software program to drive around the robot. This way the ease of use and choke points will be analysed giving an idea of the improvements necessary for the next version.

4. Ideation

4.1 Stakeholder Analysis

The stakeholders of the KUKA iDo platform can be divided into lecturers, researchers, students, and technicians.

The lecturers might use the KUKA platform in class to teach students about using a demonstration or they can let students use the platform during their projects. The lecturers should be able to answer questions the students might have that are not clear in the documentation.

Researchers can use the KUKA platform for their research projects, they should be able to utilize the platform in its simplest form and modify it for more complex behaviour and research.

Students are using the KUKA platform directly to implement the robot into their projects or research. They should be able to use the platform without any extra assistance.

The technicians are the people responsible for keeping the KUKA in shape. This consists of repairing things that might break during use and keeping the batteries charged and healthy.

4.1.1 Bob Schradenberg

An interview with Bob Schradenberg from the Human Media Interaction group was done to talk about his ideas for the applications of the KUKA platform and to discuss the possibilities of using the KUKA platform in the course Human-Robot Communication [Appendix A]. The question was if a complex robot like the KUKA could be applied here or if the architecture is too complicated. Bob was afraid the KUKA architecture would be too complicated in its current state to be used by students in a project that only lasts a few weeks. They currently use simple Arduino build kits that students can modularly extend with wheels, a face, etc. He noted that teaching ROS to students during this project would probably take too long to be feasible. Furthermore, the amount of KUKA is fairly limited, since only a few KUKAs could be used during the course, students have to work in big groups which is not ideal. He stated that a useful addition to the course could be a demonstration of robot communication at the start of the course. Some basic behaviour could be programmed into the KUKA that shows different ways to communicate actions to humans using movements, lights, and sounds. Similar programs can be used during demonstrations for events. Next to this, he saw the KUKA as an ideal platform for the thesis of a bachelor's or master's student. A limited number of KUKAs would not be a problem here since not many students would use them at the same time. The following thing discussed was the storage and maintenance of currently used robots, this is interesting since the batteries used by the KUKA robots are quite expensive and should not be neglected. He said there was not a clear system currently used to monitor robots, their batteries and other maintenance operations. This gave me the idea that with the launch of the Robotics Centre, there is a possibility to unite all available robots for research on the University in some sort of

platform on the web. This platform could show every robot, its battery condition, availability for research, charging specifications, needed software, people responsible for the robots, problems noticed by the previous users, needed maintenance, etc. This way people that are looking for a robot for a specific research area could easily find a robot that fits their needs and the people to contact. The last discussion item was the use of ROS 1.0 or ROS 2.0, this was brief, but Bob recommended ROS 2.0 since the support of ROS 1.0 will stop soon. This needs to be researched more since there is much more documentation of ROS 1.0 and certain libraries don't exist yet for ROS 2.0. However, for future-proofing of the platform, ROS 2.0 would be the best solution.

4.1.2 Edwin Dertien

An interview was conducted with Edwin Dertien, the responsible person for the course Social Robot Design [Appendix B]. This course might be best suited for using the KUKA iDo platform since students use all kinds of robots to prototype expressions of robots during their own design scenarios. These students sometimes design their own hard- and software, but these are never extremely complicated due to the time constraints. This is a similar problem Bob Schradenberg raised for the course Human-Robot Communication. There is a need for prototypes that can be used in a few hours. An interesting topic Edwin raised is that the robots should be reusable by students the following year. This means all the software should be reset to its default state. The inclusion of a reset button to match the basic software explained in the documentation is necessary to remove the need for a complicated reset protocol.

Another topic discussed was battery management, they currently use small battery packs that can easily be reused and recharged by students themselves. For bigger, complicated robots there is currently no protocol on how to keep the batteries healthy and charged. This touches upon the topic discussed with Bob Schradenberg for the need of a uniform robot protocol that keeps the robots in check and shows battery health etc.

Edwin would like to use more high-level robots during the course that need less programming for complex behaviour. He noted the Pepper and NAO robot that can for example take some steps and wave at someone using a flow chart software. The downside of such a platform is that it limits modular expansion. If someone wants to connect more hardware this would be hard to link to this flow chart program.

The last topic discussed was the form a tutorial should have. Edwin recommended a maximum content tutorial instead of a minimum content tutorial. A minimum content tutorial gives the most basic problem someone would want to solve using the KUKA. A maximum content tutorial explains very extensive possibilities of the platform using many different aspects. This gives the user the ability to remove things they don't need and still have examples of things that they need themselves. Next to this maximum content tutorial, a clean template should be supplied with the bare minimum to get everything running and get people started on their own projects.

The needed expressions the KUKA needs to be useful during the course Social Robot Design is movement, sound, light, some sort of head, and possibly a screen to display other information.

Previous projects done during the course include drones for the police, a tour guide robot and virtual dolphins for entertainment. The tour guide robot is a great example of where the KUKA platform can be used. A similar application of the platform could be a teaching assistant, a restaurant robot, and a transportation robot.

4.2 Preliminary Requirements

Using the information gathered in the research and stakeholder interviews the following requirements can be made using the MoSCoW framework.

4.2.1 Must Have

These requirements are extremely important to make the KUKA platform useful and should be implemented as soon as possible.

Convenient Maintenance

The robot will be used by many people over multiple years. This means the hardware should be stable and easy to use. The batteries should be easily rechargeable and safe to use. They should be able to sit on a shelf for a year and still be fine.

Clear Documentation

The KUKA platform should be usable without anyone's help. This means it should have clear documentation on how to use it.

Easy driving controls

The basic functionality of the KUKA is being able to drive it around. Many research experiments can be conducted using just this use case. This means a user should be able to drive the KUKA robot without any prior knowledge and without programming experience.

Software Template

There will be a lot of boilerplate code to get the KUKA working and communicating with all its components. To make sure no one has to reinvent the wheel there should be a clean template to get started with a project as quickly as possible.

Extensive Tutorial

To teach a user which functionalities the KUKA has and how to use them, there should be a guide on how to interface all the sensors, drive the motors via the onboard computer and read any data a user might need.

4.2.2 Should Have

These requirements add significant value to the platform but are not vital for users to start developing their own projects.

Modular Hardware

Since the KUKA platform might be used during demonstrations in events and in advanced research the hardware should be extremely reliable. If a piece of hardware breaks or needs maintenance it should be possible to swap out the control system for a new one to eliminate the downtime. This means the control box should function as a black box which does not need any altering for most use cases.

Demonstration Examples

The KUKA platform might be used in courses such as Social Robot Design and Human-Robot Communication during demonstrations to explain different behaviour and communication patterns to students. To enable lecturers to use it without much setup time there should be some examples preprogrammed on how a robot like the KUKA might communicate different needs, error messages and other signals. These demonstrations should fit the requirements of the course and be a useful addition to the curriculum.

4.2.3 Could Have

These requirements have a small impact if left out or are too time-consuming to implement during this design cycle.

User-friendly Programming Environment

Programming a robot using complicated frameworks like ROS (Robot Operating System) takes a lot of time to get comfortable with. To enable users without prior programming experience to develop new behaviours using rapid prototyping there should be a low-code programming environment. This makes the KUKA a platform that can be used by anyone for research, even if it is just a single experiment where the user does not have a lot of time to program complex behaviours.

Reset Option

The KUKA will be used by different researchers in different periods. To make sure the guide and tutorial stay relevant the software of the KUKA should be consistent when starting a new project. This requires some sort of reset option that reverts the KUKA back to 'factory settings'.

4.2.4 Will Not Have

These requirements are outside the scope of the current design cycle and will not be implemented in the near future.

Transparent Design

When the KUKA is able to operate fully autonomously and make decisions on its own there should be a system that displays the information the platform is using to operate and why it is making certain decisions. This makes it much easier to debug the system when anything goes wrong. Furthermore, in the future, the KUKA might use large-language models or other types of AI to extend its functionality even more. These models often operate like a black box which makes them hard to use for repeatable research experiments.

Hardware Connections

The KUKA platform might be extended with custom hardware for research that requires this. There should be available hardware connections like a second microcontroller connected via serial to the onboard computer to enable this. This hardware should then be accessible in the programming environment to extend the capabilities of the KUKA even further.

4.3 Developed Concepts

4.3.1 Multi-layer Solution

To allow all types of users to use the platform, a multi-layer platform is a possible solution. This would mean that a user that just wants to drive the robot around using a joystick should be able to pick up the remote and start immediately. A user that wants to edit some basic behaviour and interface some built-in sensors should be able to reprogram the robot in an accessible and easy interface without any prior programming experience. Lastly, a user that wants to connect their own hardware should be able to access the more advanced programming environment using actual code and extend the robot functionality. This leads to a three-layer approach where the required knowledge increases depending on the complexity of the desired behaviour. (Figure 6)

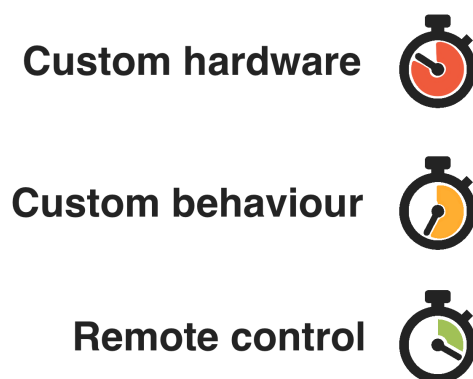


Figure 6: Multi-layer platform approach with time icons indication.

4.3.2 Motor Driver Board

The hardware of the platform should be open-source and easy to replace when needed. This requires developing a new hardware layer for controlling the robot's basic functionalities. The base of the KUKA robot is developed by an American-based company called Savioke. They develop robots for delivering products in hospitals and other environments. The hardware used by Savioke is not made to be modified, making it very difficult to add our own functionalities. This creates the need for a new PCB to connect all the controlling hardware. The hardware that should be connected includes:

- Two DC motors with a differential drive to control the wheels
- Two optical encoders to read the motor positions and enable PID control
- Three bumper sensors using Hall effect sensors
- A programmable led strip that runs around the entire base
- A speaker driver to be able to play sounds for behaviour prototyping
- A radio module to connect to the controller

The PCB should handle power distribution, two motor drivers, an audio driver, and include a microcontroller to program the needed functionalities. Where these parts are roughly located can be seen in Figure 7. The microcontroller could be an Arduino Pro Micro with a clock speed of 16 MHz. If the hardware requires more processing power a Teensy 4.0 is the next option with a clock speed of 600 MHz.

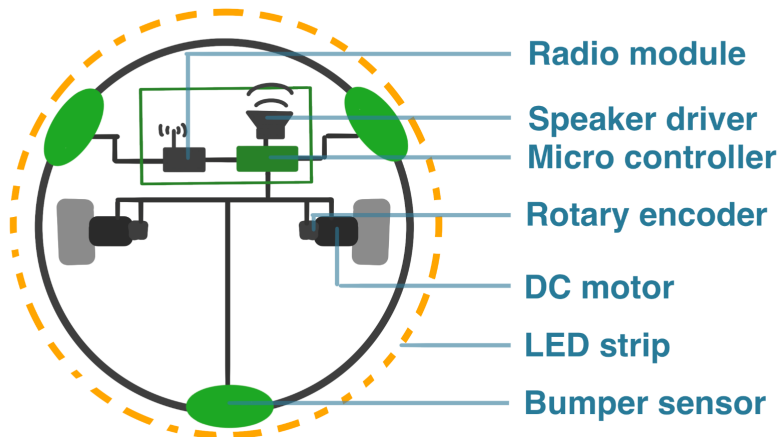


Figure 7: Basic Layout of the KUKA iDo base components.

4.3.3 Onboard Computer

The microcontroller on the base PCB will be connected to a main computer using the serial port to enable interfacing of the high-level sensors and communication with other software. The software running on this computer will be Ubuntu with the famous ROS (Robot Operating System) software. This software is the industry standard for controlling complex robotics and allows us to interface sensors like the lidar and Xbox Kinect. ROS has recently released a new version called ROS 2. A choice has to be made to use the original ROS or the new ROS 2. The original ROS has more resources and documentation available and has all the necessary libraries to interface all the sensors. The developer support for the original ROS will stop shortly, so a case for ROS 2 can be made to ensure the future-proofing of the platform.

4.3.4 Basic Controller

Researchers that just want to use the KUKA platform for driving around, displaying lights, or potentially playing sounds should be able to do so fairly easily. To enable this functionality a basic controller has to be created that works out of the box without any customization. This is also a safety feature since the robot might be controlled by more high-level behaviour programming down the line, a human-controlled controller should always be able to overwrite the movement to avoid dangerous situations. This controller can communicate to the robot via different communication protocols, the two main contenders are Bluetooth and radio. Bluetooth is extremely straightforward to set up because of existing controllers like the PS3 controller but does not have a large range, while radio has a very extensive range, but is a bit more complicated to set up. If the radio is chosen, a custom controller has to be created with some joysticks and buttons. A fully custom controller can be made, or a Wii controller can be combined with an Arduino and radio module. The Wii controller is very useful since it contains a joystick, two buttons, and a built-in gyroscope and accelerometer.

4.3.5 Battery System

The battery system is an important part of the robot. It should be reliable, easily recharged, and safe to use. The two main options are a lead-acid battery and a lithium-ion battery. The differences between these are marginal. The lead-acid battery is much cheaper than the lithium-ion battery but is heavier, less efficient, and in general less advanced than the lithium-ion battery. The lithium-ion battery are efficient and have a good weight-performance ratio but are expensive. A choice has to be made regarding which batteries to use depending on the current use of the platform. While the platform is still in development or is used for simple short-term research the lead-acid battery might be a more attractive option, while the lithium-ion battery might be a more feasible option when the platform is used for long-term research where the KUKA plays a major role.

5. Specification

5.1 Final Requirements

In the scope of the current research, only the following requirements will be met due to time constraints.

- Easy driving controls
- Software template
- Clear documentation
- Extensive tutorial

The final must-have requirement was convenient maintenance. This revolves mostly around battery management and hardware stability. Since the current state of the KUKA platform is still a prototype, the stability might not be good enough yet to allow proper research to be conducted using it. But since the platform will be developed further over the coming years by other students, the convenience of maintenance will become a more important factor.

The four chosen requirements are a starting point for the platform to find improvements for the hard- and software and enable the first user tests to be conducted.

5.2 Final Concept

The final concept of the work during this project will be focused on the KUKA driving base. This is the first part that needs to be fully developed before adding the rest of the hardware. The parts to make the base fully functional will consist of the PCB, microcontroller, remote control, computer to run the robot operating system, and finally the documentation and tutorial.

5.2.1 Hardware

Base PCB

The base PCB, which the first generation was kindly designed and supplied by the technicians of the Robotics Lab, will connect the motor drivers, motors, quadrature encoders, led ring running around the base, three hall effect bumper sensors, audio driver, two relays, and an APC220 wireless radio module. The driver for the PCB will be an Arduino Pro Micro since Edwin Dertien has a premade PCB which interfaces the Arduino with the APC220 radio module. The main PCB is designed in such a way that this smaller PCB fits right on top of it using pin connections.

Radio Controller

The basic form a user might use the platform is to drive it around themselves using a controller. For this use case, they should not need to program anything and the platform should work out of the box. A controller will be supplied to control the robot at any time from a decent distance. This controller will be the number one control, which means if any movement is sensed from the joystick, any other control programs will be ignored. This makes it easy to overtake the robot if an unsafe situation occurs. The controller will contain a microcontroller, the Arduino Pro Micro, an APC220 wireless radio frequency module, and a Wii Nunchuck as the joystick. The Nunchuck is a great solution since it contains a joystick, two buttons, an accelerometer, and a gyroscope accessible through the I2C protocol using four wires. The controller will be contained in a 3D-printed box and can be powered using a simple power bank.

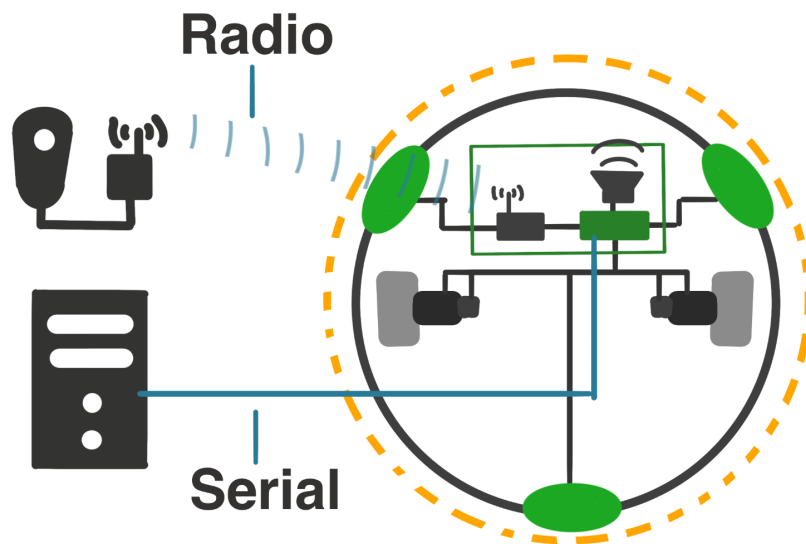


Figure 8: KUKA base connections to radio and serial computer. (Computer will be inside the KUKA)

5.2.2 Software

Base PCB

The microcontroller on the base PCB should include the following code: a function that will read the radio module to receive data from the controller. If any joystick data is received, all other control functions will be ignored. A PID loop to control the motors in a precise way which enables autonomous driving in the future. A bumper read function that measures when the robot hits anything and immediately disables the motors to prevent accidents. A function to program the LEDs to any animation or colour desired. Lastly, a library should be used to talk to the computer connected using the serial port. (rosserial for ROS, or micro-ros for ROS 2)

Computer

The computer will run Ubuntu 20.04 and the ROS (Robot Operating System) framework which is an open-source middleware suite used for robot software development. ROS recently developed a completely new version called ROS 2, which is designed to provide higher efficiency and reliability, low latency, and scalability. As Bob Schradenberg recommended [Appendix 1] and to ensure future-proofing the platform ROS 2 will be used. For the current functionality, all libraries are available but ROS 2 might cause some problems later on if users want to use advanced features that are not developed yet.

5.2.3 Documentation

With the information gathered in the stakeholder interviews, it is clear that the KUKA platform will be applied in many different fields and applications. These can consist of demonstrations during open days or lectures, in research conducted by organisations of the University or students researching specific topics during their bachelor's or master's thesis. The difference in application means that the platform should be extremely clear in use. The ins and outs of the platform should be explained in an extensive guide. This guide should include the physical capabilities of the robot, for example how to disassemble and assemble the robot, and the hardware that is connected to it. Furthermore, the guide should clearly explain the software limitations and capabilities and how to operate the system. This includes a template of all the software in its most basic form to start your own project and an extensive tutorial including most of the capabilities of the platform. This way a user can select elements from the tutorial they might need for their project.

The industry standard for documenting code based project is GitHub. This platform allows version controlled code which ensures stable code releases and a history of changes made to the codebase. Next to this it includes a wiki page per project. In this wiki a the full guide can be described which makes it easy for the researchers working on the KUKA platform next to understand the related information and steps taken.

6. Realisation

6.1 Hardware

6.1.1 Main PCB

The development of the KUKA iDo hardware platform started with the implementation of the PCB developed by the technicians at the RAM robotics lab. Since this was a fully new PCB the connections and layout of the components had to be tested thoroughly. First of all the smaller PCB board developed by Edwin Dertien which contains the Arduino Pro Micro and APC220 radio module was connected. Then the components were tested one by one using the Arduino.

Motors

The motor drivers on the main PCB connect via the smaller PCB to the Arduino. The main PCB contains two Pololu G2 High-Power Motor Drivers, which connect with two wires per driver to the Arduino, a direction and a PWM signal. Both motors were tested, but one motor seemed to only be able to run on full speed, without fine control. The reason for this is that the PCB connected the motor PWM wire to a digital pin on the Arduino that was incapable of outputting a PWM signal. This was fixed with soldering an extra wire from the Arduino to the motor control board. This just a temporary solution which can be properly fixed by developing a new PCB layout or changing the wiring from the main PCB to the smaller PCB. This is hard to do on an already soldered PCB since you have to unsolder two dozen pins at the same time.

Encoders

An encoder works by having a disk with holes spin while two sensors are measuring light through the holes. Depending on which sensor measures the light first we can determine the direction the disk spins and, and by measuring the time between the updates, we can measure the speed of the motor. The signal output is explained in Figure 9.

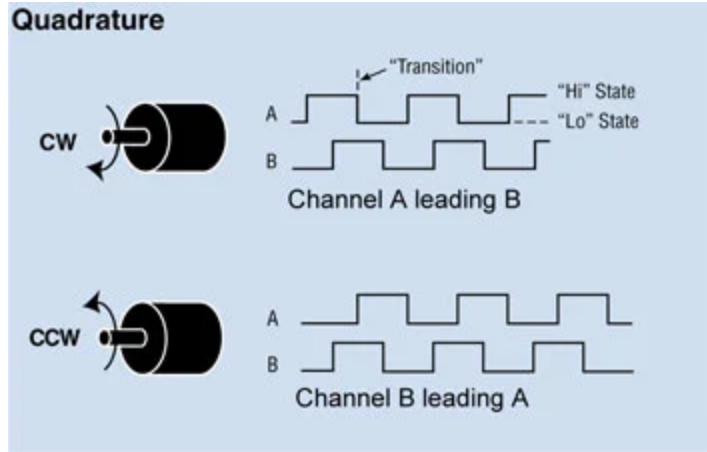
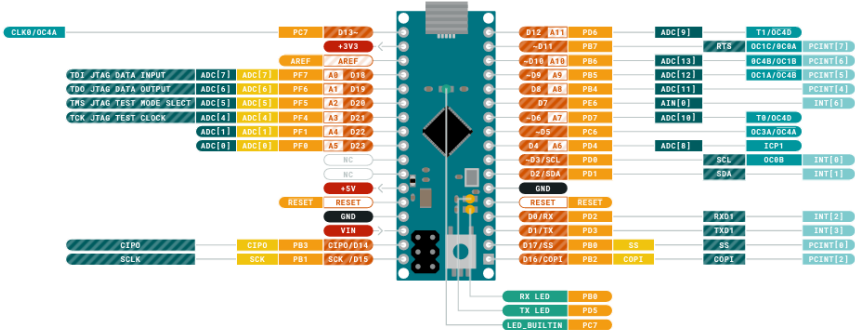


Figure 9: Signals of a quadrature encoder.

The encoders were connected to the PCB, but the wiring did not connect them to the correct pins on the Arduino that were capable of interrupts. Interrupt pins can read a signal from this pin whenever it is necessary by breaking out of the default Arduino code loop. The Arduino Micro interrupt pins can be seen in Figure 10.



Ground	Digital Pin	Analog	MAXIMUM current per I/O pin is 40mA	VIN 5-9 V input to the board.
Power	Analog Pin	Communication	MAXIMUM current per +3.3V pin is 50mA	NOTE: CIP0/COP1 have previously been referred to as MISO/MOSI
LED	Other Pin	Timer		
Internal Pin	Microcontroller's Port	Interrupt		
SWD Pin	Default	Sercom		

ARDUINO . CC
Last update: 10/11/2020

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, PO Box 1888, Mountain View, CA 94042, USA.

Figure 10: Arduino Pro Micro pin layout.

6.1.2 Radio Remote

As said before, the radio controller uses a predesigned PCB layout. Implementing this was quite straightforward. Except for the fact that the radio modules had to be synced together to be able to communicate. The APC220 modules can be put into config mode by pulling their settings pin low. After this the desired output frequency can be pushed using the serial connection. The modules frequency range is 418 MHz to 455 MHz. The chosen frequency is 444MHz.

After this the Wii Nunchuck had to be connected. This Nunchuck uses the I2C protocol to talk with the Arduino using the serial clock pin (SCL) and the serial data pin (SDA). Since an aftermarket Nunchuck was used, the pin layout was not the same as a normal Wii Nunchuck. But after some trial and error the remote was working reliable. A 3D printed case was printed to protect the components during use.

6.2 Software

6.2.1 Arduino

The Arduino Micro has the task of controlling all the components in the robot base. The most important components are the connection with the radio remote and controlling the wheels using PI(D).

Remote control

The remote control was interfaced with the Dynamixel Reader library. This library provides a process data function that can be configured yourself. The function receives three values, an xSetpoint, ySetpoint, and buttons. The x and y are an integer value from 0 to 255, and the button is a value of 0 to 3. 0 for no buttons pressed, 1 for the Z button pressed, 2 for the C button pressed and 3 for both buttons pressed at the same time. Later on these buttons can be used for certain animations or sounds. A small deadzone was implemented to limit the sensitivity of the controller and prevent the robot from moving when not intended. When needed these values can be extended with the gyroscope and accelerator data available from the Nunchuck.

Motor control

The motors are controlled using PID controllers, which need the wheel position to provide accurate speed control. This was done using the two quadrature encoders which send an interrupt signal to the Arduino whenever they sense a rotation. An encoder is connected with two wires, one of the wires is connected to an interrupt pin on the Arduino, which has the capability to stop all other code on the Arduino and prioritize the callback function of the interrupt pin. The interrupt function looks like this:

```

void readEncoder_L() {
  if (digitalRead(leftENB) == LOW) {
    posTicks_L--;
  } else {
    posTicks_L++;
  }
}
}

```

When the interrupt pin measures a signal, the function is called and the other pin is read using a digital read. Depending on the state of this pin we can determine the direction of the wheel and the position variable is updated accordingly. Using this position ticks variable we can calculate the velocity of the wheel using a delta time variable (the time between each loop) and a wheel constant calculated by Cliff ten Barge when he worked on the KUKA robot [15].

In his work he mentions that the speed is calculated by differentiating the position variable, resulting in noise in the signal when the Arduino reaches its demand. A full PID differentiates the signal again which can lead to even more noise. Because of this he made the decision to use a PI controller to eliminate the extra noise. The differential controller is normally to prevent overshoot but since the robot is meant to move steady and slowly his decision to use just a PI controller was also used here. This turned out to work perfectly fine after retuning the controller.

One major issue that occurred is that the encoders installed by KUKA are of exceptional quality and precision. So much that at a speed of about two kilometers per hour they have a combined update rate of 17,000 ticks per second. Every tick fires an interrupt in the Arduino code, which limits the amount of 'normal' loop time. When the robot moves faster than two kilometers per hour the interrupts fire so often that the normal code does not loop at all and the wheels get stuck in a full speed forward motion. For the sake of time constraints the decision was made to limit the robot speed to two kilometers per hour (3 rad/s) which works fine for the current research purposes. But in the future the microcontroller should get upgraded to a faster clockspeed. A good candidate would be the Teensy 4.0, which has a 600MHz clockspeed in comparison to the Arduino's 16MHz, while being a similar size. The smaller PCB that connects the microcontroller to the main PCB should then be redesigned to accommodate the new form factor.

6.2.2 ROS 2

ROS 2 was installed on an Ubuntu 20.04 virtual machine for prototyping since the KUKA has an Ubuntu computer in its final design. It is very useful to provide people working with the KUKA an environment that they can experiment in like a virtual machine to learn about the platform without the chance of creating irreversible damage to the normal KUKA software.

Serial communication

The idea was to use a library called Micro-ROS, which is the ROS 2 equivalent of ROS 1's Rosserial. The Micro-ROS library is a new addition to the ROS 2 system and is not yet fully tested on all microcontrollers. It turned out that Micro-ROS is not supported yet for the Arduino Pro Micro board, which seemed to force us to upgrade the microcontroller straight away. As stated before, a Teensy 4.0 would solve the clockspeed issue and perhaps also the library support issue. The Teensy should technically be supported, but when testing the system the Micro-ROS library did not compile since it is missing crucial library functions in the C compiler available. Fixing this would involve modifying the compiler toolchain used by Teensyduino, which can be risky and may result in other compatibility issues. Thus two other options were finding another compatible microcontroller, or switching back to ROS 1 since it is a more stable, tested environment.

To ensure future proofing the platform the decision was made to stick with ROS 2 and use a more simple custom serial protocol. The benefit of the Micro-ROS library is that it functions as a node in the ROS network. But to simulate a similar behavior a custom ROS node was developed that enables two way data transfer between the ROS computer and Arduino microcontroller.

ROS is based around nodes and topics. Nodes can subscribe and publish to topics to share data. The serial communication node consists of a subscription which other nodes can share data to. When the serial communication node receives this data it will write it over the serial port to the Arduino. Normally, this data can only flow one direction since it will interfere if data is also sent from the Arduino to ROS. To fix this problem the node has a 10 Hz callback function which sends the string 'RDY_DATA' to the Arduino ten times a second. When it sends this a boolean 'waiting_for_data' is set to true to disable the data flow towards the Arduino. The Arduino will receive this 'RDY_DATA' string and sends data back to the ROS computer, in this case the wheel speed and position. When ROS receives this data the normal data flow is enabled again and the cycle continues. This is explained in Figure 11.

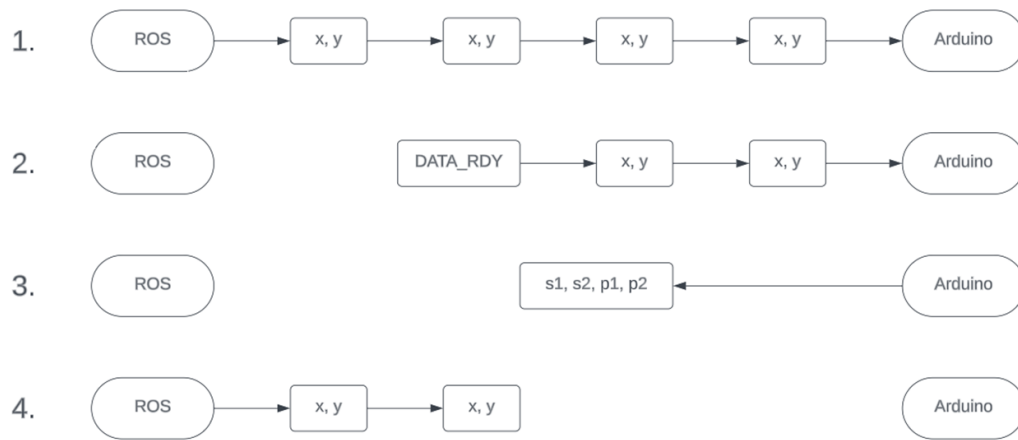


Figure 11: Two way data serial protocol.

Joystick implementation

The easiest way to test the ROS system and develop a guide to teach people the features of the KUKA is to use a PS3 controller connected to the ROS network that sends messages using the serial communication node to control the robot. This might seem a bit cumbersome since we already have a radio controller. But this joystick controller is just the start of a wide range of control systems possible. In future research the same control network can be used to utilize the KUKA's lidar sensor and develop full autonomous driving.

This means the basic ROS 2 network consists of three nodes. A serial communication node for the Arduino communication, a joystick node that interfaces with the PS3 controller and a joystick to serial node that takes the joystick data and publishes it to the the serial communication node. The full system is visualised in Figure 12.

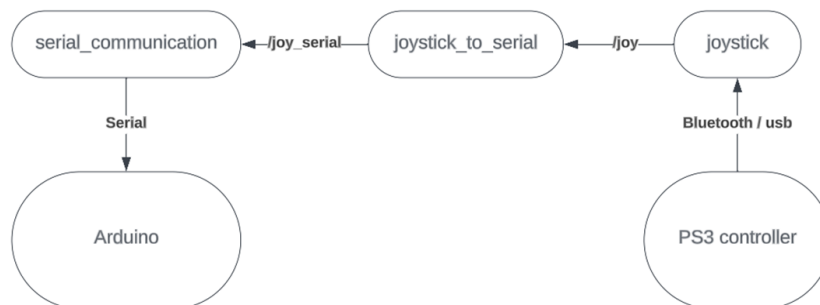


Figure 12: ROS Network containing PS3 controller and serial communication.

6.3 Documentation

The code was first created on a local machine for easy testing, but after a first stable version of both the Arduino code for the base and radio controller and the ROS 2 code on the Ubuntu virtual machine, it was pushed to an open source GitHub repository [16]. After this a guide was written describing all the steps needed to get the Ubuntu virtual machine and ROS2 code working on the computer of future researchers working with the KUKA iDo platform. The guide includes:

- Home
An introduction on what the purpose of the guide is, what the target audience is, and the the project objective.
- 1. Getting Started
A full guide on how to get an Ubuntu virtual machine running on a windows computer using Oracle VM VirtualBox. Including the initial setup for the ROS2 environment.
- 2. ROS Basics
A quick introduction of the core ROS elements the user needs to understand to follow the rest of the guide. Next to this a first hands-on practise guide on how to run the example talker and listener nodes that show how ROS nodes can communicate.
- 3. Serial Communiation Arduino
An explanation of what serial communication entails. A guide on how to setup a ROS2 workspace, and an explanation on how to create a package. Lastly a serial communication with an example Arduino is created that increments a value and sends it to the ROS node.
- 4. PS3 Controller Testing
A PS3 controller is connected to the Virtual Machine and the joystick and button data is printed to the terminal using the built in joystick node.
- 5. Combining Joystick with Serial
This guide modifies the serial communication node to make the serial data flow from ROS to Arduino. A joystick to serial package is created to read the joystick data and send it to the serial communication node. Lastly, the KUKA base is connected via usb and controlled using the joystick.
- 6. Plotting Wheel Data
In this guide the final version of the ROS2 code is pulled from the GitHub page to ensure

all nodes should work as intended. In this version the serial communication node is updated with a protocol that allows two way data transfer. The choice was made to let the user pull this code from GitHub because the data protocol is quite complicated to explain and goes outside of the scope of this beginners guide. In this guide the KUKA is controlled the same way with the PS3 controller, but this time the wheel velocity and position data is plotted using a tool called RQT plot.

7. Evaluation and Future Work

7.1 Hardware

7.1.1 Arduino

The connection of the encoders and motors with the Arduino Micro via the PCB was not immediately successful because of an oversight in the PCB design. The tracks that when from the right motors PMW control signal to the Arduino was connected to a pin that is not PWM compatible. This meant the motor could only stand still (low) or spin at full speed (high). A temporary fix was created by soldering a wire from the motor controller PWM to an Arduino pin that was capable of outputting an analog signal. In a future PCB design the track should be rewired to a different Arduino pin.

The Arduino Microcontroller turned out to severely limit the performance of the system. The choice was made to connect the motor encoders with the Arduino using interrupts. This way the most precise speed and velocity calculations can be made. While testing the motors at a higher speed they would become stuck without the ability to slow, other than resetting or shutting down the microcontroller. After some research the cause of this was that the encoders give a combined update rate of 17,000 ticks per second around 3 rad/s. Every tick the Arduino loop is interrupted, and the clockspeed of 16 Mhz was not capable of handling more updates. This limited the top speed of the motors to about 3 rad/s, or 2 km/h for the robot. To fix this problem a new microcontroller should be used in future versions. Options for this can for example be an ESP32 with a clock speed of around 160 Mhz to 240 Mhz (10/15 times faster), or a Teensy 4.0 with a clockspeed of 600Mhz (~38 times faster).

The two above stated issues can both be solved by redesigning the small PCB that only contains the microcontroller and APC220 radio module.

Due to time constraints the bumpers sensors and LED strip were not connected to the microcontroller during this design iteration. This should be quite a straightforward task and can easily be implemented by the future researchers continuing with the KUKA platform. This would also give them an introduction to the inner workings of the PCB controlling the base of the KUKA iDo.

7.1.2 Ubuntu

Currently all testing of the ROS2 network was conducted using a virtual machine on the researchers personal computer. In the future it would be wise to transfer this logic to a native

Ubuntu mini computer in the body of the KUKA. This would simplify the system and eliminate the need for complicated installation of a virtual machine on future researchers computers. It should be research if this computer should be interfaced using a direct monitor, keyboard, and mouse input, or alternatively by connecting to this computer using and SSH bridge.

Next to this, the rest of the KUKA sensors should be connected and interfaced with the ROS2 network. Example guides on how to use these sensors for a useful application should be created and added to the GitHub wiki repository. Firstly the lidar should be connected, this allows the SLAM (Simultaneous Localisation And Mapping) algorithm to be implemented. This allows the robot to construct a map of the environment, and with clever programming, the robot can autonomously navigate the environment. A perfect example of this could be with the KUKA robot that has a (touch) screen on the front. A map of the room could be created and displayed by the KUKA and a user could tap on the screen which location they would like the KUKA to move. The robot could then move to the specified location and reconstruct a new map of the room simultaneously. This would be a great example of the possibilities using a lidar sensor. Combined with the lidar the distance sensors should be used to detect objects at a higher height. The lidar is a ground level and just using this data might result in the robot trying to drive through a table. Secondly the Xbox Kinect sensor should be implemented. This camera is able to detect objects or body motions gestures which can be used to control the robot platform [18]. Lately the different add on 'heads' of the KUKA should be interfaced. This includes the camera head, the light bulb head, the nintendo switch, and the coffee machine.

7.1.3 Battery

Currently a left over lead-acid battery is used to test the system. These work great for short term research or quick testing, but are a less ideal options for long term or serious research. If the platform will be used for a long term research where it has to drive for extended periods of time, the batteries should be upgraded to large lithium ion versions. When this is the case it would be wise to create a battery charge protocol and tracking system. Since the previous batteries were destroyed by not charging them for a long time.

7.2 Software

The current software sends commands from ROS to Arduino using x and y coordinates like a joystick. When other control methods are implemented this should be converted to speed and angle of the robot. This is the industry standard of controlling these types of robots and makes it easier implement movement algorithms. Secondly, the current communication between ROS and Arduino is a custom serial protocol, this works fine for the current applications, but if future tests show that the data rate should be increased it might be wise to switch to the ROS-Micro library. This allows the microcontroller to become part of the ROS2 network and will make it easier for

multiple nodes to communicate with the KUKA base. Lastly, the software for the other KUKA parts and sensors should be written and implemented in the current system.

7.3 Documentation

The initial goal of my research was to create a platform for researchers to easily pick up the robot and create the behaviour they need for their specific research. Since the scope of the thesis shifted more towards the technical side of the KUKA iDo, the decision was made to still create a user friendly guide for following researcher to continue with the KUKA iDo platform.

When testing the guide initially and installing the whole system on a new computer, the virtual machine setup alone took more time than the rest of the guide. For this reason this step was skipped during user testing the guide and the recommendation is made to experiment with different ways to get an Ubuntu instance running. This can be done using a native Ubuntu computer, or by utilizing a Docker container.

The guide was tested by two students, one with a little bit of programming experience and the other without programming experience at all. Before following the guide they were asked what if they were interested in robotics and if they understood how they worked. They followed the guide without any help, except when a fault in the guide was discovered or they forgot something important that was required to move forward. During the guide they were asked to give pointers on steps they found unclear and a description of exactly what they thought was unclear. After the test they were asked to explain their general thought of the guide and if a system like this worked as they expected.

7.3.1 Participant with Programming Experience

The student with a little bit of programming experience explained that they were interested in robotics, but never really worked with them before. They did think they had a bit of an idea on how it works, since they worked with Arduino's before. The students with programming experience managed to finish the whole guide in approximately 30 minutes. Some general pointers they gave after following the guide were that felt like it was mostly running commands in the terminal and not enough explanation of the core concepts behind the system. Furthermore they said the guide was extremely clear and the steps were small and easy to follow.

7.3.2 Participant without Programming Experience

The students with no programming experience at all struggled a lot more with the guide. The total finishing time was 60 minutes, but they did need a lot more help during the process. The observation was made that this students had a lesser knowledge of general computer concepts

needed to work with a virtual machine and the Ubuntu operating system. Knowledge they did not have were for example the concept of a terminal and how it can be used to modify directories, create files, and run programs. Another tendency that was noticed that this students never saved a file after editing it, even though it is clearly stated in the guide. A hypothesis for this is that the overwhelming amount of new information makes them a worse reader causing them to forget steps or not understand them correctly. An interesting line of research would be to figure out how a guide can best explain a step based process while ensuring that the user follows each step correctly. This is hard to do when the guide and the users system are not connected in any way. A solution for this would be to create a virtual web based environment where the guide and an example operating system are both available on the same page. This allows the guide to check wether a user had done something correctly. This concept is used more often on web based guides where users learn programming concepts like CodeCademy [17]. It might be a challenge to get ROS to run and compile in a web based environment, but it would probably allow for a better learning environment for new robot developers.

7.3.3 General Improvements for Wiki

In general the following improvements per chapter were suggested for the wiki guide by observing the participants.

- 2. ROS Basics
 - Explain how pasting code in a terminal requires the user to right click and click ‘paste’, or to press ‘control + shift + v’.
 - Explain the concepts of a shell startup script and a setup file.
- 3. Serial Communication Arduino
 - Explain how the command ‘cd’ controls directory navigation in a terminal.
 - Explain better what the correct output of a command should be in the console.
 - Explain why you need to unplug and replug usb devices after connecting them to the virtual machine.
 - PS3 controller sometimes goes into bluetooth search mode and disconnects from the USB cable. Explain how to fix this issue.
- 5. Combining Joystick with Serial
 - Explain why the changes in the serial communication node are necessary and describe what the specific blocks of code do.
 - Describe in a more elaborate way how to create a new file using the terminal.
- 6. Plotting Wheel Data
 - Highlight in a very clear way when a new terminal should be opened.

8. Conclusion

The goal of this research was to find a practical use for the KUKA iDo service robots, specifically in enhancing their accessibility and applicability in the field of human-robot interaction. Literature showed similar research platforms and their techniques of making behaviour prototyping accesible. The two main contenders were code block or flow chart type interfaces.

With these considerations in mind, the scope and requirements of the research were defined. The primary focus was on rebuilding the platform, specifically developing new control hardware for the base of the KUKA robot. This primarily involved controlling the motors using PID control and establishing communication between the robot, a radio controller, and a central control computer. During the research the main adopted control software was ROS2, the industry standard in controlling advanced robots. A basic control network in ROS2 was developed to control the KUKA robot via serial communication from the central control computer running Ubuntu.

The ROS2 software was thoroughly documented using a GitHub wiki repository. The steps were evaluated by having two students with different levels of programming experience follow the guide. It showed that the setup time of this software was marginally decreased, which should lead to a decrease in learning time for future researchers working with the KUKA platform. The GitHub repository is publicly accessible which means future researchers can fork it or gain access by contacting the original researchers to further develop and extend the code base and wiki guide.

The research is relevant since the KUKA robots contain a vast amount of high quality hardware, which can be used to teach students or to allow for research using the robots. Finishing the robot platform can be a learning experience for students wanting to conduct their thesis in the realm of robotics. This gives the benefit of giving students the opportunity to learn more about robotics using a high quality robot platform while the platform in not in its finished form.

Once the platform is in a more mature stage of development, it can be used in human-robot communication research and hopefully contributing in a more synergistic relationship between humans and robots in the future.

Appendix A - Transcript Interview Bob Schradenberg

Geert Mol

Jij geeft de course Human Road Communication. Want daar leren jullie Mensen ook om, zeg maar gedrag in te programmeren bij de robots. Wat voor platform gebruiken jullie daarvoor?

Bob Schradenburg

Gebruiken nu Arduino toolkit voor robotic toolkits. Daarmee kunnen studenten zelf hun robot een beetje vormgeven. Vervolgens kunnen ze daar bijvoorbeeld een Shell omheen bouwen. Zo kunnen ze eigenlijk een eigen robot snel bouwen die wel wat dingetjes kan zijn. Nog wel vrij gelimiteerd, vaak in ook in. Hoe expressief ze kunnen.

Geert Mol

Want het zijn rijdende robots of meer gezicht of wat wat bijvoorbeeld.

Bob Schradenburg

Ja, het ligt eraan wat je erop zet, dus Het is modulair dus. De de basis is. Een Arduino en daar kan je van alles dan aanhaken, dus je kan een gezichtje opzetten met LED's je kan wielen eraan zetten bijvoorbeeld. Mobiel aanzetten bijvoorbeeld. En zo hebben we nog een aantal andere actuators.

Geert Mol

Wat hebben jullie zegmaar, modules gebouwd of zo, die je. In elkaar klikt of.

Bob Schradenburg

Ja, die kan je dan in elkaar klikken en dan. Zelf programmeren ook.

Geert Mol

Ja oke cool dus.

Bob Schradenburg

En, Dat is dan de elektronica en dan zelf kan je de shell omheen maken die dan het uiterlijk vandaag over bepaald en hand daarvan. Is het doel dus om te gaan kijken van. Ook, hoe kan je met zo'n robot communiceren zodat Mensen het begrijpen

Geert Mol

ja ja. En die dingen gebruiken ook accu's of hebben jullie dat niet?

Bob Schradenburg

Powerbanks, geloof ik dat gebruikt hiervoor.

Geert Mol

En dan puur voor het onderhouden van die dingen. Doe jij dat of hebben jullie technici die dan die accu's opgeladen houden of hoe werkt dat?

Bob Schradenburg

Studenten, kunnen we dit lenen van zitten die apparatuur en zij zijn het volgens aan het gebruiken. Daar kan ook het.

Bob Schradenburg

Zo, dat moeten ze zelf verzorgen en aan het eind van het project levert weer in. En Edwin Dertien, die heeft dat in beheer.

Geert Mol

Want er zitten nog onderhoud buiten de cours over die robot. Of is dat helemaal niet nodig?

Bob Schradenburg

Dat weet ik niet. Dat zou je Edwin moeten vragen.

Geert Mol

De Mensen die bij jullie komen, hebben die al ervaring over programmeren of helemaal nog niet soms, of wel.

Bob Schradenburg

Ja vraag wel dat ze ervaringen met programmeren.

Geert Mol

Ook met Arduino ja oke, dus je hoeft niet de basis. Dingen uit te leggen.

Bob Schradenburg

Nee, We hebben wel tutorials daarvoor, dus tijdens het werkcollege kunnen ze daarmee aan de Gang ook en met die tutorials zouden ze het ook moeten kunnen vragen. Wel dat ze wel enige ervaring daarin.

Geert Mol

En dat vervormen hebben die tutorials tekst of video of.

Bob Schradenburg

Tekst zijn het.

Geert Mol

Dus dat is gewoon een soort PDF je uitgeprint, of.

Bob Schradenburg

We hebben een wiki ervan gemaakt en er staan een aantal links in. Soms maken we zelf ligt een beetje aan welke tutorial. Je pakt, soms hebben die die zelf schrijven, soms linken die gewoon naar resources.

Geert Mol

Ja precies.

Bob Schradenburg

zodat je dat niet Opnieuw een wiel uit hoeven te vinden, ja.

Geert Mol

Ja voor de KUKA ben ik bezig om die gewoon meer de basislaag helemaal opnieuw te bouwen, dus We hebben nu een nieuwe PCB ontworpen met hardware en nu ben. Ik aan het connecten, en dan komen we aan het schrijven om alle bestaande hardware te interfacen. Daar ben. Ik nu een beetje halverwege mee, zie jij voor de Kuka, want mijn idee is een beetje om, want We gaan er met Ross connecten en dan dat je daar wat ingewikkelder behaviour je voor in kan programmeren. Zie jij voor de koeka, zou je denken van oh, dat zou wel nuttig zijn voor het vak Human robot Communication.

Bob Schradenburg

Op dit moment denk ik dat die vrij lastig is om om te gebruiken, Omdat je vrij snel een vrij grote robot architectuur nodig hebt om te. Te kunnen gebruiken. En daar hebben we denk ik niet de tijd

voor om daar echt. Dieper op ingaan, dus dan zouden we bijvoorbeeld al Ross moeten. Uitleggen hoe dat werkt. Dat punt het vak is onderdeel van de Robotica master en dus wel een van de eerdere dingen. Geloof ik die ze leren hoe, hoe ROS werkt. Maar het zal waarschijnlijk te laat. Zijn voor dit. Dat is, want We moeten eigenlijk gelijk al beginnen met het programmeren. In elkaar zetten en dan het programmeren van de robot.

Geert Mol

Ja, Het is echt een later pas ROS.

Bob Schradenburg

Ja of gedurende het vak, Maar dat zou dan. Daarnaast is het doel Natuurlijk niet van het vak dat ze een hele complexe robot per se hoeft te gebruiken, Maar het. Is meer dat ze nadenken over hoe ze. Met gemuteerde mogelijkheden kunnen communiceren. Ja, wat zijn de methodes die. Je daarvoor aan doet?

Geert Mol

Ja precies, dus eigenlijk is juist de kracht dat gelimiteerde.

Bob Schradenburg

Ja en dan zie ik op dit moment nog even niet de meerwaarde in. Van een geavanceerde robot waar we er? Ook niet heel veel van hebben.

Geert Mol

Nee, Dat is Natuurlijk wel een nadeel, want Ik weet niet meer hoeveel kits je hebt van die. Andere waarschijnlijk wel op.

Bob Schradenburg

Ja ja ja, een stuk of 10 volledige kids hebben, geloof ik.

Geert Mol

Ja, dat merk ik dat het de grootste pitfall een beetje is dat Iedereen zegt, van ja, We hebben zoveel studenten. Je kan eigenlijk niks met 3 robots, of Je moet echt in grote groepen of maar een deel hiermee Laten werken. Maar dat is Natuurlijk niet echt nuttig.

Bob Schradenburg

Ja jij refurbished is nu eentje.

Geert Mol

Ja, dus ik begin nu gewoon met een en dan wat we. Daarvan leren kunnen. We toepassen op de rest? Ja. Want de koek is Natuurlijk wel best groot en indrukwekkend zou je Misschien wel beter kunnen testen wat 1 grote robot voor impact heeft. Qua behaviour zou het qua bijvoorbeeld een demonstratiedag of een lecture dat Mensen er gewoon mee kunnen experimenteren? Zo dat leuk of nuttig kunnen zijn.

Bob Schradenburg

Dus dat soort dingen zijn wel nuttig, denk ik. Voor voor een lecture is het Misschien meer dat je het gewoon laat zien en van wat hij dan bijvoorbeeld kan, net zoals bij een een demo dag en daar zijn we ook bezig mee bezig om dat om daarvoor in te gaan zetten. Verder kan ik me ook voorstellen dat het interessant platform is voor masterstudent, bijvoorbeeld om een voor bachelorstudenten op een afstuderen opdracht. Te doen Als ik bijvoorbeeld een student van mij het laatst. Die heeft ook de robot gebruikt dat we Alleen nodig om via smaakte, stelt Alleen de Remote Control ervan gebruikt, maar daar is denk ik ook wel een interessant platform voor.

Geert Mol

Dus meer voor onderzoek denk je is beter?

Bob Schradenburg

Ja, dus, dan hebben ze ook de. Tijd om er wat dieper op in te gaan en daar echt wat? Ja iets gaat maken wat wat zou kunnen werken en wat complexer.

Geert Mol

Is en wat als er nou iemand een soort platform voor de koeka ontwerpt, waardoor je eigenlijk heel weinig los kennis nodig hebt en best wel snel met dat simpele code iets? Van gedrag zo in kunnen programmeren, zou het dan meer nuttig zijn voor een lektuur? Denk je of zou het dan nog steeds te weinig robots zijn nog steeds te geavanceerd zijn?

Bob Schradenburg

Nou ja, voor een lecture voordat je uiteindelijk daar bent, Omdat moet je ook vrijwel. Bij uitgelegd dus dan? Een selectie al vrij snel over wat je wel kan doen is.

Bob Schradenburg

Hoe het wel kan, gebruiken ze bijvoorbeeld de robot een week gebruikt, dus dat je. We hebben ook eerder gedaan, dan heb je bijvoorbeeld de Pepper robot en nog een paar andere robots. Dan

kan de studenten een groepje studenten kunnen dan een robot kiezen en leg het mee. Gaan ze dan een week? Bijvoorbeeld programmeren of twee week. Dan is het Misschien weer haalbaar is. Dan is inderdaad die. De hoeveelheid van die robots probleem?

Geert Mol

Ja, want het zou zou het een optie zijn om bijvoorbeeld maar een select groepje of Mensen die het echt heel erg leuk vinden met een robot te Laten werken. En Er is met een simpelere robot zou het kunnen.

Bob Schradenburg

Voor het master vak bedoel je? Ik denk dat we het liever hetzelfde houden om.

Geert Mol

Voor de ta's Enzo?

Bob Schradenburg

Ook, maar ook om de je krijgt wel extra complexiteit weer daarnaast en er wordt ook moeilijker voor ons om te beoordelen.

Geert Mol

Dat snap ik.

Bob Schradenburg

Jij dan mee omgaan?

Geert Mol

Want je hebt nog, geeft nog meer vakken met waar je robots in gebruikt, of niet?

Bob Schradenburg

Niet ja, maar die coördineer ik verder niet dus, maar Er zijn wel andere vakken waar we ook robots gebruiken, maar ook studenten wat We moeten. Programmeren aan een robot, dan nemen ze dan vaak een week of 1,5 vorm voor.

Geert Mol

Want wat ik me nog afvraag of er meer op het platforms die we op het moment hebben, die bijvoorbeeld wat meer geavanceerde technologie en ook grotere accu's gebruiken, hoe we dat onderhouden? Soort van, want zo'n accu kan niet een half jaar in de kast staan, terwijl die bijvoorbeeld is daar een soort systeem voor, weet je dat?

Bob Schradenburg

Nee, dat weet ik zo niet. Want we gebruiken vrij veel robots Natuurlijk, maar hoe het met? Die accu gaat weet ik niet. Maar die blijft gewoon lang doen. Ik neem bijvoorbeeld aan nou en de Pepper die we dan hebben. Dan zou je eigenlijk met Daniel moeten praten.

Geert Mol

Wie is dat?

Bob Schradenburg

Alleen Daniel Davison, Hij is de lab manager, oké. Hij is nu op ouderschapsverlof. Ik wist even niet wanneer hij terug is.

Geert Mol

Ik zal eens kijken.

Bob Schradenburg

Dan maar in principe managed hij alle dingen.

Geert Mol

Ja voor jouw cours zou het Alleen voor demonstratie in het echt. Zijn denk ik voor de lectures of voor de studenten zelf wat minder. Ken jij cursus waar het Misschien wel nuttig zijn die toevallig voor mastervakken met heel weinig studenten of waar ze in groepjes van 10 werken ofzo, waar ik Misschien eventueel wel.

Bob Schradenburg

Groepjes van 10 wel groepjes van 3 of 4, dus We hebben. Foundations of indirect of Technology is een intro cursus en daar hebben we vaak ook een. 1 1 2 weken dat ze met een robot bezig gaan. En daar kunnen ze dan ook uit meerdere robots vaak kiezen volgens mij en zo heb ik het vroeger voor. Hem gegeven en Interactive Technology. Verder andere vakken zijn capita selecta, waar een student dan iets mee met een robot een onderzoek kan vormgeven. Hebben we nog? Ja, hoe heet het vak? Iets van Research topics kunnen ze ook een onderzoek zelf gaan uitvoeren.

Bob Schradenburg

Ja, ik zit nog een eindje denken, want. Volgens mij heeft het iets van gewoon van itech project of iets dergelijks en dan. Gaan de studenten in een groepje aan een onderzoek?

Geert Mol

En wat voor wat voor onderzoeken kan ik bij dit soort vakken voorstellen? Wat ze dan met KUKA zouden doen? Heb je wat voorbeelden bijvoorbeeld?

Bob Schradenburg

Nou bijvoorbeeld grapje wat ik vroeger heb begeleid was dat ze een sociale robot gebruikten. Voor autistische kinderen en dat ze daar een interactie mee hadden gemaakt, geprogrammeerd. Vervolgens hebben ze die ook getest op. Op een school.

Geert Mol

Dat doet ze een robot? Dan gaan we praten bewegen?

Bob Schradenburg

Ja, ik geloof van alles had een doel is dan om iets van sociale vaardigheden aan te leren. Dus inderdaad die die interactie moest, is dan programmeren, dus het gedrag van wat doet hij? Hoe doet u dat? Ja, zegt hij, wanneer zegt?

Geert Mol

Hij dat? Ja, dat jij dat? Zag grappig en vorige keer hadden we het ook over dat de Unie voor de Europese Unie iets in ziekenhuizen voor. Hoe noem je zo'n? Ding in UAV maakt ofzo, ja?

Bob Schradenburg

Klopt, Dat is een onderzoek spreekt waar ik aan. Werk Europees project. Die student die de KUKA heeft gebruikt, deed ook onderzoek binnen dat project. We hebben daar een prototype nu van van de robot. Die is op dit moment in Portugal. Dus We hebben niet altijd. Toegang en de KUKA iDo die. Die lijkt er wel een beetje op qua uiterlijk. Dus in die zin gebruiken we die dan als proxy.

Geert Mol

Dat is te doen of gebruiken jullie dat al of wil je dat gaan doen?

Bob Schradenburg

Ja idealiter gebruiken, we gebruiken gewoon echt prototype Natuurlijk, Maar dat is voor de grote onderzoeken proberen we die dan hierheen te halen, maar bijvoorbeeld van master studenten waar dat niet altijd haalbaar is dan. Kijken we bijvoorbeeld of ze die?

Geert Mol

Dat gebeurt nu nog niet of wel?

Bob Schradenburg

Nee die master studente.

Geert Mol

Oké, want wat doet zij precies voor onderzoek?

Bob Schradenburg

Zij bepaalt hoe je type geluiden gebruikt kunnen worden om een bepaalde situatie duidelijk te maken. Die geluiden, ik probeerde robot. Iets mee te communiceren? Nadat verschillende typen geluiden om. Ja in verschillende categorieën om te kijken naar hoe goed die het deden.

Geert Mol

Ja ja, ja, dat kan ik me herinneren. Want stel, we zouden dat soort dingen voor zo'n project in koeka vaker willen gebruiken. Zijn er op de Unie is er een soort systeem om te kijken wat voor robots er aanwezig zijn en wat je eventueel zou kunnen gebruiken. Of is dat helemaal niet een ding?

Bob Schradenburg

Volgens mij niet een UT breed systeem, We hebben wel voor onszelf. Voor ons lab hebben we dat wel?

Geert Mol

Ja en Dat is dan de HMI groep.

Bob Schradenburg

Zeg maar ja. De Interaction lab heet het.

Geert Mol

Zou jij het nut in zien om een uni breed robot catalogoog is daar hebben we van beschikbare robots die over Research zou kunnen gebruiken.

Bob Schradenburg

Het lijkt me wel wat complexer dan Alleen het systeem op te zetten, want Je moet ook Mensen hebben die weten hoe ze moeten programmeren. Zomaar een robot die. Wil je niet altijd uitlenen, Omdat je ze soms. Voor bepaald TYPE onderzoek snel nodig hebt. Of dat ze extreem moeilijk en duur zijn om te gebruiken dus.

Geert Mol

Maar dat zijn Natuurlijk in kunnen stellen dat je die robots dan lokt of niet erop zet. Of het instellen van hier moet je die kennis voor hebben of Alleen maar voor dat soort studenten beschikbaar.

Bob Schradenburg

Ik denk dat het wel nuttig is om te zien. Van welke lab, welke TYPE robots hebben? En wie de contactpersoon daarvoor zijn.

Geert Mol

Dat als je er iets mee zou willen doen, je weet wie Je moet bereiken. Dat zou wel interessant zijn. Want je hebt Natuurlijk dit, je hebt de Ram. Heb je nog meer robot labs die veel robots beschikbaar zouden hebben voor dat soort ding?

Bob Schradenburg

Jij ligt eraan wat? Je onder robots schaaft dus. Tech net even bijvoorbeeld volgens mij ook een ja meer medische robot. Precies neering hebben we ook werk ook aan allerlei robot stukken.

Geert Mol

Ja, ja, Dat is Natuurlijk leuk. Misschien met het Robotic Center dat alles wat meer verbonden wordt, dat je ook. Zoiets op zou kunnen? Zetten, wat allemaal die labs wat meer te verbinden. Ja had ik verder nog iets? Nee, Ik denk op zich dat ik alles wel heb stel. De cuca is, Ik weet niet of dat nog in mijn afstudeeropdracht afkomt, maar die is een keer klaar voor een demonstratie of zo. Wie zou dat? Dan kunnen deze module nog Omdat in jouw vak een keer te proberen of iets dergelijks of wat voor demonstraties zou je dan nuttig vinden?

Bob Schradenburg

In september pas weer.

Geert Mol

Stel, je zou de koeka dan voor een lecture als demonstratie de eerste twee lecture krijgen, wat voor dit zij leuk vinden om te Laten zien, zeg maar.

Bob Schradenburg

Dat hij bijvoorbeeld bepaald. Ja, hoe die dingen kan communiceren en dan niet zozeer met gesprek. Totaal Maar dat je andere modaliteiten gebruikt, bijvoorbeeld beweging of licht, of.

Geert Mol

Ja een insinuaties die die jij dan neerzet, zoals of hij rijdt ergens tegen een muur aan of staat een persoon recht voor hem ineens of ja, moet je antwoorden sensor hoe die dan aangeeft van hé, dit doet hij. En zou jij dan de. Hoe die robot reageert de inprogrammeren of zou je dat voorgeprogrammeerd willen hebben of zou je wat opties willen hebben?

Bob Schradenburg

Voor een demonstratie is het handig dat het voorgeprogrammeerd is, denk ik. Sowieso Omdat we dit dit platform ook willen gebruiken voor demonstraties, zou je bijvoorbeeld dat kunnen denken dat je daar een deel van Van laat zien? Wat hij daar zou kunnen?

Geert Mol

Voorbeeld maar je wilt er Natuurlijk wel wat goede voorbeelden hebben, want je wil niet een hele slechte communicatie. Of nou ja, dat kan ook nuttig zijn voor die studenten van. Hé, Dit is. Niet helemaal goed, dat zou dat veel uitmaken of het echt goede communicatie of of denk je van nou iets? Is ook oké.

Bob Schradenburg

Als je een demonstratie Misschien beide interessant dus. Dat je eerst? Dat zie je dan. Want opties meer experimentele mogelijkheden die we Misschien wel moeilijkste importeren en een andere die. Echt van weten dat hij een duidelijk is dat je dan verschil kan kan zien.

Geert Mol

Ja ze is eigenlijk een soort interface willen hebben waarmee je kan instellen van. Nou, nu wil ik dat hij dit doet voor dat of juist Dit is een goeie en Dat is slechte gewoontes simpel opties, zoiets of zo? Dat zijn interessant. Cool ja dan ben ik wel tevreden eigenlijk. Ja ja, heb jij nog vragen?

Bob Schradenburg

Kun je wat vertellen over wat je precies nu met de KUKA doet van wat je aan het upgraden bent, want je zet daar ROS. Dit draait al op geloof ik.

Geert Mol

Ik toch eens de oude versie, Wij hebben nu eigenlijk de hardware uitgetrokken, want het oude platform wat erin zat was gemaakt van door een savio oké. Dat zou bedrijf in Amerika Alleen al die printplaten en software. Daar kan je niks in doen, dan kan je niet bij kan je niet kosten customizen, dus de technici van de Ram hebben nu een eigen PCB ontworpen waar we met Arduino kunnen interfacen. met de robot en dan maken we een basis controller eigenlijk met een windjack die gewoon direct met die Arduino communiceert en daarop sluiten we een computer aan die dan met serial met Arduino ook communiceert, waarmee je dus eventueel complexere dingen met ros met de leider dat soort dingen in kan stellen en. Alle lampjes enzo. Dus dan voor complexe dingen heb je dan een ROS met met alles wat je wilt en dan Als je gewoon wil rondrijden, kan je meteen de Wii controller gebruiken Zonder dat hij verder iets hoeft te doen eigenlijk. En ja, dat maak ik nu een beetje dan. Als studenten dat willen gebruiken, maak ik nu wat tutorials om dat dan in te stellen en ros te gebruiken voor toekomstige onderzoeken.

Bob Schradenburg

Ja, Dat is dan ROS 2?

Geert Mol

Ja, dat ben ik nog niet helemaal over uit, Ik weet. Niet, wat heb jij daar aanrijdingen over van?

Bob Schradenburg

Na twee gaan, want een wordt is niet meer ondersteund, dat wordt niet. Meer ondersteund wordt.

Geert Mol

Oké ja ja.

Bob Schradenburg

En Als je gaat kijken naar de levensduur van dit plaatje. En dan ook weer wat langer is, dan zou ik. Wel zeggen na ros twee.

Geert Mol

Want ik zat een beetje te googlen toevallig deze week ben ik daar begonnen en er was 1, 4
Natuurlijk heel veel documentatie over ROS twee, niet zoveel nog, dus Dat is wel weer een
nadeel.

Bob Schradenburg

Maar sommige dingen, sommige Libris heb je ook niet voor als twee, geloof ik. Maar Omdat de
support stopt van ros een, zou ik wel. Echt aanraden naar ROS 2 gaan.

Geert Mol

heb je zelf ook wel eens gebruikt verder.

Bob Schradenburg

Maar ik weet wel, een beetje van hoe het werkt, en We zijn nu. Of onze eigen modules die we
vaak gebruiken. Die zijn nu ook wel eens twee compact.

Geert Mol

Grond, dus Dat is net echt voor als ze willen combineren of iets in die richting.

Bob Schradenburg

Ja voor die demonstrator willen we. Dat ook gaan combineren? Dus dan zou. Het makkelijker
Als het al opgelost twee draait dat we dat zo drink kunnen.

Geert Mol

Dat is wel slim. Daar ga ik mee aan de slag. Dankjewel succes.

Appendix B - Transcript Interview Edwin Dertien

Geert Mol

Yes, Het gaat gewoon over de het vak sociale robot design. Jij geeft ben dan vooral benieuwd wat jullie er precies doen, want ik zet even de osiris pagina ook door te lezen dat jullie bepaalde tooling, frameworks en platforms zijn zo gebruiken en dat jullie daarmee expressies van robots en zo en andere puppeteerin opdrachten maken. Als we het goed begrijpen.

Edwin Dertien

Bijna helemaal studenten moeten zelf een casus formuleren, een een gebruiks casus rondom een robot platform nog gebruiken een bepaalde robot als startpunt. Waarom de studenten of op dit moment waar omheen de studenten van de casus? We hebben In het verleden wel ook wel gewerkt met geformuleerde casussen door een bedrijf waar we dan een robot platform of of een een robot onderwerp bij zoeken, werkt ook dit nu meteen omgekeerd. Dat is ook wel een interessante, vooral Omdat studenten dan. Gedwongen worden of gedwongen vooral uitgenodigd te worden om een gebruikersgroep te kiezen die dicht bij hunzelf ligt, dus vaak gewoon zichzelf als gebruiker of direct iets of iemand uit je omgeving. Wat het vaak wat directer maakt om een casus te bedenken en die ook daadwerkelijk vorm te geven dan We gaan een hypothetische casus voor een hypothetische gebruikersgroep ver weg op een andere planeet waar we toch nooit bij kunnen. Dus Dat is een beetje het uitgangspunt de studenten. Die starten met een. hardwareplatform en het idee van een casus ontwikkelen daarmee scenario's. Gaan vervolgens door de lenzen embodiment, expressie en gedrag kijken naar wat ze dan als ontwerper met zo'n robot precies allemaal moeten kunnen. Daar komt soms ook wel een een stuk hardware ontwerp en software ontwerp bij kijken. Maar we merken dat dat best lastig is binnen de korte context. Van zijn vak. Dank je voordat je goed en wel onderweg bent met een idee en een casus ben je eigenlijk alweer 4 weken verder en voordat je dan ook nog eens de programmeer tools of hardware tools onder de knie hebt om daadwerkelijk iets te bouwen, dan ben je vaak al weer. Wordt de tijd heen die je nog zou willen benutten om 1 keer 1 experiment of een prototype of eens. Een test te doen? Dat maakt het dus dus best. Lastig dus de Er is. Om op die manier te kunnen? Werken wel een behoefte aan aan robot Als hardware of bijvoorbeeld. Prototypes waar je eigenlijk gewoon binnen een dag of een paar uur. Mee uit de voeten kan En, Dat is nog best lastig, want Het is niet Alleen Maar het het snel kunnen opstarten om van hardware van berijdende stuk robot hardware de ruimte binnen en dan werkt het. Daarna moet het vervolgens die robot hardware ook lange tijd of stilstaan of in andere projecten gebruikt kunnen worden en vervolgens weer heel snel in een soort. Startklare staat dus terug worden gebracht. Zo 3 zetten op soort soort grote grote reset, maar dan wel met volle batterijen. Nou en dat? Dat maakt het allemaal best best lastig, heel veel van bijvoorbeeld de ontwikkelomgevingen van Van robot platforms die zitten accounts vast en accounts zit hem vaak aan een abonnement

of een tijdelijke registratie of iets anders vast. Dus voordat je van een bepaalde ontwikkelomgeving die bij een bepaalde robot ooit de accounts de versies de software hebt gedownload. De tools hebt geïnstalleerd en en het bemanning band, dan zit hem er soms alweer 3 weken mailing met een fabrikant. Heen en weer om vervolgens te ontdekken dat het weer niet binnen de educatieve licenties past en wel of niet kan. Nou ja etcetera dat dat soort geëmmer heb je. Vrij vaak bij vrij.

Geert Mol

Zet op tijd is gewoon echt te lang.

Edwin Dertien

Fors dus een hele lange z op tijd. Het is dan ook vaak zo dat battery management is is lastig, want ja, In de tussentijd liggen die robots ergens op een plank en bekommert zich meestal Niemand rond. Maar ja, Dat is dan vervelend om na weer een volgende periode zeg maar weer 3/4 jaar mijn robot van de plank te halen en er komen ik dat of de batterijen helemaal dood zijn of niet In de goede oplaadt toestand, achtergelaten of wat dan ook dus ja dan dan kun je alsnog nog niks met de hardware of kun je Alleen maar met de hardware werken met een flinke. Ja nou dat dus de de en dan nog de laatste de hoeveelheid tijd die je daadwerkelijk kosten om je bepaalde tools eigen te maken. Die is nog los van Van puur, het het al dan niet open en makkelijk distribueer ware karakter, maar gewoon puur de complexiteit en die die dingen die maken. Het eigenlijk heel lastig Om op die manier binnen het korte tijdsbestek van zo'n vak daadwerkelijk iets nuttigs met zo'n robot.

Geert Mol

Te gaan doen, want het. Hebben jullie wel Natuurlijk Als ik het. Zo hoor wel eens gedaan.

Edwin Dertien

Met robots hebben we zeker zeker wel geprobeerd. En wat er dan dan vaak gebeurt, is dat een robot al dan niet gepuppeteert of maar gewoon hard gecodeerd geprogrammeerd. Een bepaalde routine doet die wordt vervolgens ingekleed in een filmpje en als video prototype de wereld in geholpen en daar wordt een kwestie nergens over gedaan, dus op zich best best te doen kun je best wel wat wat kennis mee achterhalen, Maar het zijn Natuurlijk veel leuker zijn om om. Om in ieder geval echt zelf een een studie te kunnen doen, al is het Alleen maar voor de vorm en met met wijze van spreken Jaargenoot jes gewoon uit te proberen, maar wel zo een zo een studie daadwerkelijk zo een experiment op te kunnen zetten met de fysieke robot en daar een leuke AB test mee te doen. Of gewoon nou ja, wat we ook veel doen is, is met de simulatie door acteurs dus dus een robot met een gesimuleerde gebruiker en interactie Laten doen om daar vervolgens dingen uit te uit te kunne. Nou, daarvoor is het soms genoeg dat een robot Alleen maar

gepuppeteert wordt, dus Dat is vaak toch waarop teruggevallen wordt. Dus of dat een robot een vastgestelde routines frequentie en een aantal bewegingen gestuurde bewegingen even achter elkaar doet. Wat zeg maar in een filmpje goed gaat soms met een goed getrainde acteur. Eigenlijk naartoe zou willen, is dat je ook echt een stuk robot gedrag en meer high level robot gedrag kan. Ja kan programmeren wat meer wat verder gaat dan Alleen maar van nou ja, als bewegingssensor x iets iets detecteert, dan moet je. Gedrag y doen. Maar het liefst nog met nou ja, wat voor? Elke andere sociale modaliteiten is een robot inderdaad. Een een nette, een net stukje taalmodel kan gebruiken en een gesprekje kan hebben en dat je dat dialoog kan ondersteunen met bijvoorbeeld de juiste bewegingen of met kleurtjes, of nou ja, andere embodiment of of.

Geert Mol

Wat dan ook nog nooit gelukt met autonome robots?

Edwin Dertien

Nee, Dat is is eigenlijk met echt echte autonomie. Tenzij de autonomie zelf of de. Nou ja ja, dat lukt wel met autonome robots. Alleen dat lukt met hele simpele autonome robots bij heel veel extra gedrag of tributes op projecteert. Je kunt namelijk best wel met een met een domme zeehond doen alsof je heel slim of met een domme robotkat doen, alsof je hele heel heel slim onderzoek doet. Sterker nog, Als je erbij zegt dat die robotkat heel erg slim is, dan wordt het meestal het gedrag ook als heel erg slim geïnterpreteerd. Dus Dat is. Prima dus in in die zin is dat dan een autonoom robot, want je item hier en dan maakt die geluidje A poging om daar dan maakt die geluidje B en Dat is allemaal autonoom zelf niks aan te doen. Alleen ja, de relevantie daar dan weer. Van voor onderzoek. Is weer is weer. Kleiner en dus een echt platform waarmee je wat meer high level dialoog. Toch en op bijvoorbeeld dialoog interactie kan doen, gecombineerd met zoals beweging rondrijden, een tourguide robot of wat dan ook die je als ware als kant en klaar platform krijgt. Waar je eigenlijk? Nou ja, voor je eigen gesprek, gespreksonderwerp of gespreksmodellen hierin of je eigen nieuwe set bewegingen die de robot doet om Mensen met. Mee te lokken en en ga met die banaan dat dat kan niet.

Geert Mol

Dat zou eigenlijk het ideale zijn dat Mensen zelf hun bewegingspatroon kunnen ontwerpen dan zo'n dure.

Edwin Dertien

Dit is, er zit wel echt echt een een soort soort disjunctie, een soort scheiding tussen dingetjes, simpele sensor interactie de Zense think act, maar dan nog simplistisch eendimensionaal niveau. Ik heb hier een aanraaksensor en dan maken we geluidje A of gaat het lampje B of motortje C aan dat dat dat soort interactie nou, daar hebben we genoeg platforms voor zijn studenten goed

genoeg voor? Ongerust, We hebben interessante robot platforms die auto of the box het standaardgedrag. Nou ja, leuk genoeg doet op camera en Als je de goede stukjes eruit knipt, kun je het best wel. Gebruiken voor video prototyping. We hebben een paar platforms die zich redelijk Laten puppeteeren, maar dan is er realtime ook vaak nog een dingetje, dus Dat is dan weer meer peteren dat we een aantal bewegingen of bewegingspatronen erin zetten. Ga opnemen voorprogrammeren. Maar echt een een high level platform waar goed standaardgedrag in zit, wat daadwerkelijk ook ook makkelijk aanpasbaar is.

Geert Mol

Nee, wat zou volgens jou de basis benodigd zijn? Hele benodigdheden zijn voor standaardgedrag van iets meer high level robot.

Edwin Dertien

Nou ja, kijk, Je kunt bijvoorbeeld een een na of een of een Pepper als voorbeeld pakken. En daar zitten, bijvoorbeeld in ieder geval high level bewegingen of high level commando 's in van bijvoorbeeld ga staan en en lopen 3 passen en zwaai met je armpje of kijk, kijk naar naar de de spreker en. Ook de omgeving kon je ook wel af die je daarvoor kan gebruiken, is nou ja best ook voor niet programmeurs behapbaar. Het is de Signal Flow, dus klik gewoon een blokje en als dit dan dat en dan heb je een pijltje ertussen met een conditie en dat dat heb je snel genoeg. Iets interessants mee mee voor elkaar, dus dat maakt dat platform wel weer. Waar het niet dat de de robots zelf Natuurlijk de Peppers en met nou zelf zijn In de de embodiment en juist die bewegingen. Wel weer redelijk beperkt in wat ze kunnen en juist op dat vlak waar je Misschien qua sociaal robot onderzoek is iets anders met een andere embodiment of andere typen bewegingen. Of andere modaliteiten waaraan ze binnen toe toevoegen, iets zou willen doen. Ja, dan mis je daar. Weer de aansluitpunten voor.

Geert Mol

Ja, want hij kan eigenlijk niet meer dan rondlopen en de hand. Uitsteken een beetje.

Speaker

Nou ja die.

Edwin Dertien

Ik, je bent bent daarbij volledig beperkt tot tot tot de functionaliteit aan boord en is er geen uitbreidingsmogelijkheid. Dat is Natuurlijk ook nog wel een ander ding, dus Als je met een met een platform op de op de voeten komt en het. Nou ja, Het is. Zo open en transparant dat je er direct mee aan de slag kan Zonder moeilijke accounts. Dat soort dingen de de hoeveelheid

instructie die je nodig hebt is laagdrempelig genoeg. Dan is er Natuurlijk ook nog het punt op bijvoorbeeld elektrisch niveau of elektronisch. Waarom dat ook de interface naar dus een andere sensor of een andere input of een andere output of een ander device daarop aansluiten, dat je dat ook mogelijk?

Geert Mol

Zo in maken Als je zo een nieuwe basis platform opstelt aan limiteert dat ook weer uitbreiding, dus Dat is wel een soort afweging die we maken.

Edwin Dertien

Er is dus bij het bij. Het bij het maken van die basis is dat zeker nadenken over over uitbreidingsmogelijkheden is. Natuurlijk ligt dat niet zo zo enorm vast, want Je kunt heel veel prototypen waarbij je met gescheiden systemen werkt. We hebben een laatste leuke gedaan voor de. Worden De politie, waarbij er een een soort arm van nou ja een een robot addon bedacht moest worden voor een robot hond om sensoren in een ruimte te kunnen plaatsen. Nou ja, en waar De politie waarschijnlijk aan zat te denken, was gewoon een robotarm op een robothond. Maar spot spot ja en ze hadden dus ook spotachtige robot de Unitree ja, dat ze dus aan alle teams die daaraan meededen beschikbaar gesteld. Nou beneden ook met met een designlab studenten. Mee uiteindelijk is dat een hele leuke robot sensor plaatsen voor gemaakt op basis van een rolmaat om dat al van we are niet kan heel makkelijk uitrollen en en hoog op de goede hoogte een Center met met lijn tegen de muur aan zetten en daarbij precies de taak doen die je wil, maar niet ook nog andere, bijvoorbeeld gevaarlijke dingen zoals wapens of andere andere. Nou ja, meer meer meer tricky dingen kan gewoon. Simpele dingen doen. Toen heeft er best gekeken naar of het mogelijk was om vanuit die die robot hond interface niet spot minitree, Maar ik zal wel uit om daar vanuit dan iets aan te kunnen sturen en zo. Nou, er waren geen makkelijke, simpele open protocollen, uitgangen, tjes of andere.

Geert Mol

Wat ik niet heb uit te? Breiden op die platforms.

Edwin Dertien

Zomaar niet, niet niet simpel of niet goed gedocumenteerd.

Geert Mol

Dat zijn ook een beetje gat In de markt, achter stuk uitbreidbare platforms.

Edwin Dertien

Als je dan weg hebt, dan krijg je vaak zo een moeilijke uitbreiding met een ingewikkelde API waarbij je een soort rudimentaire driver krijgt waarbij je zelf alle drivers code moet schrijven en je hebt een uitbreiding sport. Nou ja, dan moet je hopelijk maar zoeken dat je daar iets op aangesloten krijgt. Wat nou ja, jezelf nog niet heel erg heel erg helpt. En een alternatief zou bijvoorbeeld zijn gewoon een hele simpele poort of We hebben gewoon een standaard arduino hier gewoon liggen en daar staat ook wel serial op en nou ja, je mag zelf programmeren wat je erin wil en hier hebben we een elementaire sketch en via rossio dan krijg je deze commando 's binnen en op die manier mag je hierop aansluiten. Wat je? Wil? Maar ja, dat zou.

Geert Mol

Ook kunnen dat echt de meeste studenten zeggen dat we aan kunnen. Ja, Dat is wel interessant.

Edwin Dertien

kijk als Als het het de bijvoorbeeld Als we nu uitgaan van robot met een bepaald bijvoorbeeld een een roze frame wordt erop of zo en en je hebt studenten überhaupt toch al om wegwijs te worden naar hardware uitgelegd hoe de de bottom layer met een met een Arduino wordt aangestuurd. Dan is een klein stukje tutorial uitbreiding, hé. Wil je nu zelf een nieuw rood lichtgevend hoofd opmaken? Nou dan neem je deze rally module. Je trapt je er 12 op af en hier heb je een ros serial Arduino. Die je zo aan onze template sketch voor de Remote puppeteering stopt en zo dus in software verder verwerkt in een autonoom programma.

Geert Mol

En in ros kan je een redelijk makkelijk zeggen van signaal dit daar naar die Arduino, hoe hoe gemakkelijk de student dat oppakken.

Edwin Dertien

Nou ja, dat dat, dat weet ik dat, dat weet ik niet hoe makkelijk studenten. Dat oppakken dus. Ik denk dat dat staat of valt bij de gratie van de goede voorbeeldprogramma 's en templates die je die je ze geeft.

Geert Mol

Dat de teriaal, zou ik die? Er bij me ja. En wat voor?

Edwin Dertien

En en tutorials, Dat is vaak heel vervelend tutorials, die gaan vaak uit van Van het minimum wat je nodig hebt en ga hier zelf op uitbreiden. En ik hou juist van tutorials waarbij je het maximum krijgt en ga maar schrappen wat je niet nodig hebt en dat zit ook veel meer in lijn met een CreaTe

achter. Aanpakken, dus Als je bij wijze van spreken een tutorial geeft waarbij je in in ros via een externe Arduino 3 sensor gegevens binnen labelt en als topics published en 5 Ledjes aanstuurt en nog een andere ingewikkeld ding. En dat als voorbeeld geeft, dan geef je in plaats van nou, hier heb je je void loop en je void main en en Je moet dit of dit doen en en. Goed, leuk, dan geef je Mensen hemels andere basis, dus iets geven dat al werkt en dat al redelijk top notch is wat kan worden uitgekleeft, is soms een beter beter startpunt dan nou, we geven jou gewoon de de basics en die nou ja wel of niet iets doen. Oh ja, maar dan moet je wel je eigen content.

Geert Mol

Naar toevoegen, want anders gebeurt er niks uit. Ervaring is dan beter om dan toch bijvoorbeeld drie tutorials te maken die steeds niveau omhoog gaan of een tutorial te maken waar je een soort van stap voor stap door die hele moeilijkste casus heen gaat.

Speaker

Ik denk denk dat je?

Edwin Dertien

Dat je dat je twee verschillende dingen gewoon moet doen, dat je gewoon de de, want kijk aan de hand van een ingewikkelde al helemaal ingevuld werkend prototype wat je wat je wat waarvan je breakdown doet. Dat is super lastig om dat allemaal in tekst te stoppen. Dus wat je wel wil zien, is gewoon eerst even het hele kale framework en waar horende dingen en hier heb ik gewoon een schone schone sketch en en Dit is Samen, zeg Maar de randvoorwaarden en Dit is dus standaard library die hoort er gewoon te zitten, dus Dit is wat je krijgt Als je. Kind en daarnaast een ingekleeft voorbeeld met vrij veel verschillende dingen.

Geert Mol

Alle bellen en toeters.

Edwin Dertien

Zodat je ziet hoe verschillende types toeters en. Bellen allemaal benut worden.

Geert Mol

Dus dan heb je eigenlijk twee opties, maak je.

Edwin Dertien

Ja ja nee, die die twee opties is wel fijn om te hebben. Dit ene heb je nodig als vanuit educatie gezien gewoon om je systeembeschrijving goed te doen. En de de de dus meer de de wat Laten zien en de andere is dan toch wat meer gewoon. Nee, maar hoe kan ik dan nu hier in deze sensor aan een grafiekje of deze sensor gebruiken om de robot te Laten stoppen op het moment dat de sensor te dichtbij is? Dus We hebben Ultrasoon sensor Arduino extra Arduino rossio was erop, dus de topic hebben we daar ergens in rössle of in in ros draait mijn main loop main loop, die kan ik verder goed, wie starten met deze startknop en stoppen met die knop? En Als ik hem heb gestart dan draait die programmaatje laat dit grafiekje. En gaat rondrijden met deze twee motoren. Nou, ik zie dat daar naar dat republics stop ik dat daar de snelheden naartoe worden geschreven. Oke In de robot. Die heeft zich daarop geabonneerd. Oké, en dan zeg ik up en dan hè, gaat die en dan heb je een basis applicatie en dan kunnen Mensen wel zelf weer verder gaan.

Geert Mol

Uitbreiden die die uitgebreide tutorial dan bijvoorbeeld leuk zijn om zoiets te pakken als India herkenning van Hé, ik doe dit met mijn hand herkent hij dat qua image en dan. Stop je stopteken. Of gaat het dan weer te ver vind je?

Edwin Dertien

Nee, dat kun je, kun je pakken dat op niveau dat je wil, zeker als Als je je robot platform met vrij veel ingewikkelde sensoren. Ja, ik zitten, dan is het wel heel fijn dat dat de juiste libraries de juiste tools en dus ook de juiste voorbeelden voor die die die systemen Er zijn, dus even voor jou jou een opdracht is dat enorm Beyond the scope. Ik zou dat lekker houden bij een een simpele Arduino en nog niet proberen volledig transparant te Slam met IM You en odem. En de realsense camera's en de lidar en alles tegelijkertijd te doen. Helemaal goed, ik moet. Die namelijk Misschien een keer master opdracht gedaan worden. En Als je nou eerst paar vakken zoals je robot Communication, een robot Operation Frameworks en robot software development volgt, Nou dan heb je volgens mij voldoende spul om om dit als een hele interessante master te kunnen doen. Als je hem dan al? Niet helemaal zat. Bent hebt aan de andere kant, Je kunt wel de tijd die je er dan nu hebt in geïnvesteerd In het platform leren kennen en per ongeluk zelfde hardware ervoor ontworpen hebben wel allemaal gewoon netjes dubbel gebruiken.

Geert Mol

Ja, dat geeft wel een goede blik op ieder geval waar ik op moet focussen bij zijn tutorial en ook dat moet toepassen. Verder hadden we het even over onderhoud en batterij management en zo, want daar zijn we nu wel robots die dat nodig hebben. Want hoe wordt dat onderhandeld? Wordt dat nu gedaan, doet de. Technici, dat doe jij dat?

Edwin Dertien

Ja nu nu nu is het soms Als ik eraan denk dat ik even denk, oh, We gaan binnenkort weer laat ik maar eens even alle batterijen eruit halen. Oh, dit wordt zomervakantie. Misschien moeten alle batterijtjes eventjes uit dingen worden gehaald en en gecheckt en dan niet. Te laden dus, Maar ik denk dat daar dat dat heel veel hapsnap is, dat het bij de meeste opstellingen in een lab vaak ligt bij de desbetreffende onderzoeker of studenten die ermee bezig is. Ik vind.

Geert Mol

Het is niet een duidelijk systeem in ieder geval.

Edwin Dertien

Nee zeker niet een duidelijk systeem. Daardoor konden dus ook gebeuren dat er hier een aantal KUKAs terecht waren gekomen. Die zijn ergens in storage gekomen. Ja daar je zou kunnen zeggen, daar had dan een technicus direct naar moeten kijken ze direct uit moeten pakken, batterijen eruit moeten lepelen en alle laders bij robots moeten aansluiten en een cycle moeten ontwerpen voor voor de de de. Hoe zegt dat de battery management? Maar ja, dat gebeurt Natuurlijk.

Geert Mol

Ja nee, Dat is wel leuk, want het sluit heel erg aan waar ik het met Bob over had. Dat het geeft wel. Een idee van waar de Unie op het moment Misschien op uit zou kunnen breiden, is dat robot platform waar alle robots op staan, zou je ook eens batterijen, soorten en lading protocollen. En wie daar wie dat moet doen en Misschien daar zelfs een verantwoordelijke of twee of zo voor aanstellen die alle robots lading organiseren.

Edwin Dertien

Nee, je zou Als je dit dit een interessant onderwerp In het verlengde van Bob, zou je dan ook met Daniel nog eens even bij bij H niet kunnen. Want zij hebben meer nog dan dan bij je bij RAM kant en klare robots op batterijen. Wij hebben hier eigenlijk niet zo heel veel drones. Nou ja, er gaan altijd battery packs in en die zitten wel in een speciale ladekast. En dat wordt door de onderzoekers gedaan. Maar we hebben niet zoveel van die naos en Peppers en andere kant en klare platforms. En Ik denk dat de meeste platforms daarbij HMI gewoon kant en klaar zijn en dus van batterijen en lader. Voorzien, ja, en Ik weet niet hoe en wanneer hij ze later.

Geert Mol

Toekomst met de Robotics master zou dat Misschien uitgebreid worden. Dit soort dingen en robots? Dan zijn we slim dat goed.

Edwin Dertien

Nou goed, dan is het, ja, dan is het ook een hele duidelijke en interessante vraag van hé, waar stal je deze robots als er zo meteen goed genoeg zijn? En, waar zet je hun onderhouds laders neer en wie doet dat zo nu en dan? En hoe vaak?

Geert Mol

Want ik gebruik Natuurlijk bij de koeka nu zo een goedkope lood accu. Ja, nou, Ik heb nu gewoon. Aan een uitspaart. Maar ik zat al even te kijken en Als we die accu's meer vervangen, kost zo'n € 1.000 per accu minimaal 800 extra accu.

Edwin Dertien

Nou de echte accu's. Ja, de echte accu's dat we?

Geert Mol

Die willen gaan doen. Ja, Dat is een duur grapje dus stel je zou dat willen doen, dan moet er Natuurlijk wel wat opgezet worden. Dan zijn die over. Een half jaar weer.

Edwin Dertien

Nou ja, maar goed, dan moet je ook gewoon overwegen of je daar een doel voor hebt en waarschijnlijk is, is het doel van zo nu en dan de met een studentenproject je iets iets mee gaan gaan Laten rijden. Is dat niet doel genoeg om zon zon uitgaven te verantwoorden? Je zou wel kunnen zeggen van hè op het moment dat er langdurige experimenten en er worden een twee koelkas ingezet voor een experiment op Schiphol. Wat een paar weken gaat duren. Nou, dan is het een mooie mogelijkheid om weer even in voor goede batterijen en een goed managementsysteem en een goed plan te gaan investeren. Maar niet nu.

Geert Mol

Denk ik ook dat er Misschien een optie zou kunnen zijn om dan die accu dan niet per se voor de koek uit te kopen, maar meer voor hè? Deze accu is beschikbaar.

Geert Mol

Voor alle soorten. Robots die je zou willen bouwen, dat zou.

Edwin Dertien

Wel mooiste zou zijn inderdaad wanneer alle robots op dezelfde types accu's werken. Ik heb al een tijdje ben ik daar nu wel voor bezig voor bijvoorbeeld de wearable tech set die we die We hebben voor om om wat dingen te prototypen heb ik uiteindelijk gekozen om hele simpele. Van die power bankjes te nemen als batterij die je makkelijk kan opladen en ook voor 101 andere dingen gebruikt, dus die worden nu ook voor 101 andere dingen gebruikt. Dus Ik ben ze altijd kwijt. Maar hebben wel eens voor dat dat zou worden gebruikt en dat iemand zijn dan nog een keer oplaadt Omdat ze. Voor iets anders worden gebruikt?

Speaker

Worden sterk maken.

Edwin Dertien

Ja en en ze zijn. In ieder geval makkelijk genoeg te vervangen wanneer je met zo een setje in je hebt geen bouwwerk. Nou ja, haal je even een andere powerbank dus dus dat in plaats van een set, een normale accu's of een set specifieke accu's of een setje penlites of wat dan ook is wel een, heeft wel weer een voordeel.

Geert Mol

Voor de kuga, wat zou je dan nou ja, in mijn scoop aanraden? Gewoon, Ik heb met die.

Edwin Dertien

Leuk denk ik dat met die lood acties werken nog best best makkelijk is, want ook de performance daarvan is niet niet niet superslecht en je hebt niet een mega gewichtsprobleem, dus dat scheelt. Ze zijn in die zin vrij vriendelijk in laden en worden niet snel overladen zijn wel op een gegeven moment Natuurlijk. Als je ze te lang laat liggen, gewoon dood. Dus dat blijft, dus Het is wel zaak om ze zo nu en dan even allemaal eruit te pakken op een rijtje te zetten en even de laadklep aan te sluiten. Maar dat is wel een keer te doen, ja. Duidelijk dus binnen binnen, dat zeg Maar de jaarcyclus waarin ze nu gebruikt worden, zou ik denken dat dat. Simpele loodaccu's de meest kosteneffectieve manier zij. En en dat uiteindelijk beter doen dan dan.

Geert Mol

Dure lithium Ion Dat is uiteindelijk Misschien wel het mooiste, Maar het is wel.

Edwin Dertien

Heel duur, nou ja, kijk Als je aan de andere. Kant weer de. 22 koffiezetapparaat wil gaan gebruiken en andere grote toeters en bellen. Kijk bij dat soort toeters en bellen. Dan is het wel zaak om om voldoende vermogen te hebben. Ja, Dat is dan.

Geert Mol

Weer wel dat terzijde nog een vraag over de expressies die studenten Misschien dan in zo'n robot zouden willen inbouwen. Neem je bijvoorbeeld het KUKAs platform wat? Voor expressies zouden studenten allemaal moeten kunnen gebruiken als als tools om dus een bepaalde gedrag of iets dergelijks.

Edwin Dertien

Ik denk dus dus hele goede controle over over de rijweg ging besturing. Een systeem voor geluid, een systeem voor licht lichtsignalen, licht licht. Lettering omschreven dat soort dingen, maar wat mee kan vervolgens toch een Misschien een uitbreiding naar een een hoofd of een ander meer duidelijkere communicator. Dat ding, Ik denk dat het het ontwerp leent zich niet zo om nou direct twee explosieve armen aan te schroeven, dus dat zou je kunnen noemen die zieke niet zo snel van komen. En en rare dingen waarbij er een tentakel of een arm of een blok of een ding afkomt. Nou ja, Misschien. Wij is een variant met een met een soort soort orp, een lichtgevende bol. Nou ja, Dat is een een grappige toevoeging. Inderdaad, Ik had het al over licht en.

Geert Mol

Ja, want nu heb. Je die die ledstrip, want de basis zeg maar.

Edwin Dertien

Hoort hij ook een soort vis ja In de achterplaats eventueel uit te breiden naar iets iets wat je daarvoor op op simplified een een scherm wat er op de voorkant zit.

Geert Mol

Ja, dat hebben er eentje van volgens mij.

Edwin Dertien

Ja nee, dat zijn er volgens mij twee met een VSA man, maar een met een scherm. Hebben een scherm hebben. Kijk In het voordeel van de schermen. Daar kun je Natuurlijk van alles op. Daar kun je van gezichtjes op avatars tot datastructuur tot teksten.

Speaker

Nou ja dan nog.

Geert Mol

Want je mist heel erg nu in ieder geval bij die robot bedenkingen heel erg en gezichtsuitdrukking of naar social robot te maken en Dat is een scherm makkelijk op te. Lossen met mede processing weet ik veel.

Edwin Dertien

Daar maar juist juist het doen van sociaal robot gebracht bij dingen waar niet een op wiers gezichtje op zit, vind ik ook qua onderwerpen heel geschikt en sociaal op iemand af kunnen rijden en dingen duidelijk kunnen maken in beweging. Is soms nog veel leuker dan nou ja, ik herken een gezichtsuitdrukking kje? Ik bedoel, volgens mij is is het nog steeds ongeveer 80% wat we waarnemen is is non verbaal en non verbaal is niet Alleen Maar de de gezichtsuitdrukking. Dat is een heel klein stukje van de aspect om de rest is echt allemaal beweging en houding snel. Ja, en Dat is juist wel weer super tof om dat ook te onderzoeken met zijn platteau. Ja hoe, hoe doe je de onderhandeling bij elkaar passeren In de gang hoe doe je de onderhandeling bij het aangeven van een bord soep? En dan niet de onderhandeling waarbij het expliciet wordt gemaakt, een expliciet verbaal, maar inderdaad.

Geert Mol

Of nee precies, maar een lampje of ja cool, als laatste kun je me een paar korte voorbeelden geven van een paar casussen die studenten hebben gedaan het afgelopen jaar of de jaren.

Edwin Dertien

Met je bedoelt met de de koeka?

Geert Mol

Of met het vak bij bij social robot design.

Edwin Dertien

Klopt ja even denken, bijvoorbeeld een een drone voor politie als robot, die situaties moet kunnen deëscaleren en dan als onderzoeksvraag, hè, wat is bijvoorbeeld een een hoogte waarop zo'n drone zijn dingen gaat doen?

Geert Mol

Wat wat? Hoe zou die drone iets escaleren?

Edwin Dertien

Nou ja nee juist wie.

Geert Mol

Nee deed eerst.

Edwin Dertien

De escaleren nou ja, Dat is precies dus de vraag van wat voor stemgeluid, wat laat je zo'n droom zeggen en vanaf wat voor soort hoogte doe je? Dat doe je dat inderdaad letterlijk heel erg. Vanuit de hoogte hier spreekt De politie. Ga allemaal nu liggen op de grond met je handen. Nou ja inderdaad Iedereen plat of of zijn er andere andere tactieken. Ik denk een ander soort onderzoek beetje zijn vergelijkbaar, je hebt. En, oh ja, je hebt een een gids robot in een groot gebouw of in een ziekenhuis of zo. Nou ja, hoe kan kun je non verbaal de standaard dingen van follow me of ga deze deur door of hier naar links of. Naar rechts, hoe kun je dat qua beweging doen? Ook een beweging vraag van, hoe krijg je iemand met bijvoorbeeld voor virtuele dolfijn zwemt gedrag wat iemand juist het water in lokt? Wat voor gebaren kun je daarmee maken voor voor meer voor speltherapie In het ziekenhuis als dat bedoelt We hadden? Even denken wat een lekker spitten en graven. Non verbaal gedrag voor spionerende fotolijstjes hoe laat je bij een fotolijstje wat eigenlijk? In principe ook sociale robot is, was een product van genus genus care hoe laat je vanuit dat product non verbaal zien wanneer je wakker of actief spionerende dan wel camerabeelden registreert? Of juist niet? Doen gaat het inderdaad.

Geert Mol

Wat pioner aan het fotolijstje, Dat is.

Edwin Dertien

Dat nee, je hebt een fotolijst met eigenlijk alle functionaliteit van een van een smartphone erin waarbij het Het is een fotolijstje wat bij oudere Mensen op op het Bureau staat, waarbij je iedere keer de leuke foto's op de laatste foto's van je kleinkinderen kan Laten zien. Maar aangezien het op die manier met in verbinding met de kleinkinderen staat. Kun je het ook gebruiken om nou ja, bijvoorbeeld een video connectie te maken en. Te gaan video bellen en. Die rijdt je dat direct via dat ding, wat Natuurlijk gewoon fijn maakt. Je hebt gewoon schilderijtje ergens staan staat er altijd. Het is niet de opa kan vergeten op te laden. Of niet kan. Knop die kan drukken. Nee dus dus? Je hoort zo nu, dan hoor je een belletje en dan moet je gewoon op een knopje klikken en dan heb je in een keer je kleinkinderen daarvoor. Je leeft de telefoon meisje. Maar goed, dat betekent dus dat er een een camera is en allerlei andere sensoren op zo'n fotolijstje die aanwezigheids kunnen detecteren. En dat. Dus Dat is allemaal voor de greater good, want daarmee kun je inderdaad vroeg waarschuwen wanneer opa niet uit zijn bed is gekomen of lees gevallen, of weet je niet wat dus Het is? Voor veiligheid is het heel fijn, maar ja, dan heb je dus

wel een lijstje met een camera en allerlei sensoren wat jouw dag en nacht aan monitoren. Is dus privacy is dan vaarwel. Hoe geef je dan op een interessante of of?

Geert Mol

Privacy is een probleem.

Edwin Dertien

Op een ja. Wat is dan een interessante manier om bijvoorbeeld? Verbaal daar met die privacy en vooral daarmee de interactie, de mogelijkheid of die eigenaarschap weer terug te gaan geven aan een oude persoon. Gordijntjes Natuurlijk dicht en daarna mag het. Mag het device dus niks meer geven. Maar wat gebeurt? Oh, daar ben ik weer. Nee, ja wel. Nou ja, dat dat dat soort dingen? Nou ja, dat, dat zijn zijn een paar van de Van de van de project product ideeën, dus heel erg over welke bewegingen, welke mobiliteit en wat soort embodiment. Wat voor interactie mechanieken.

Geert Mol

Van die 4 is dan voor de koeka, bijvoorbeeld Girod Natuurlijk de meest Niemand.

Edwin Dertien

Ja is is is 1 is 1 gids robot, een tour guide robot, een robot voor een robot, docent voor instructie In de klas, een robot om een restaurant robot te gaan, vast wel een keertje weer krijgen. Een robot om nou ja, gewoon bepaalde goederen weer te dispatcher en dingetjes weergeven. Ziekenhuis afval ophalen ik nooit.

Geert Mol

Het is echt een soort.

Edwin Dertien

Ezel eigenlijk, ja, nou ja, goed, nou, ja, Dit is precies wat Het is. Het is een transport robot en Als het zeg maar van A naar B kan rijden dus dus in opdracht van jouw dingen van A naar B kan brengen. Nou ja, dan krijg je een paar sociale dingen, namelijk op het moment dat je in dezelfde ruimte zit, namelijk hoe, hoe passeer je elkaar? Wat doe je als er iemand voor je gaat staan en dat soort dingen? Hoe geef je als robot Als je van Mensen afhankelijk bent voor een deel van je functionaliteit, bijvoorbeeld de lift in of de lift? Uit, hoe geef je dan? Hoe maak je je wensen op een op een subtiele manier kenbaar op zo een manier dat Mensen ook daadwerkelijk gaan gaan helpen? Hoe doe je de interactie dan wel? Onderhandeling van klepje open klepje dicht aan te geven wat wat iemand eruit moet pakken of of juist niet, dus daar zitten best wel interessante vragen In het sociale domein.

Geert Mol

Allemaal dingen vormen het scherm, dan gaat die automatisch een star doen. En zo cool, Ik denk dat ik wel genoeg informatie heb interf jij nog denkt van dit moet je nog echt weten.

Edwin Dertien

Nee, nee, Ik denk dat de meeste dingen voor onderwijscontext in ieder geval wel wel. Hebben gehad? Ja goed dankjewel, oké alsjeblieft. Nee is goed om dat te.

References

- [1] "Relay Robotics," 2023. [Online]. Available: <https://www.relayrobotics.com/>. [Accessed: 18, May, 2023].
- [2] "Nao SoftBank Robotics," 2023. [Online]. Available: <https://www.aldebaran.com/en/nao>. [Accessed: 22, May, 2023].
- [3] "Pepper SoftBank Robotics," 2023. [Online]. Available: <https://www.aldebaran.com/en/pepper>. [Accessed: 22, May, 2023].
- [4] "MiRo-E," 2023. [Online]. Available: <https://www.miro-e.com/>. [Accessed: 22, May, 2023].
- [5] Weintrop, "Block-based programming in computer science education," *Commun. ACM*, vol. 62, no. 8, pp. 22–25, Aug. 2019. doi: 10.1145/3341221.
- [6] Weintrop et al., "Evaluating CoBloX: A comparative study of robotics programming environments for adult novices," in *Proc. 2018 CHI Conf. on Human Factors in Computing Systems*, vol. 366, pp. 1–12. doi: 10.1145/3173574.3173940.
- [7] A. Topalidou-Kyniazopoulou, N. I. Spanoudakis, and M. G. Lagoudakis, "A case tool for Robot Behavior Development," 2012. Available: https://link.springer.com/chapter/10.1007/978-3-642-39250-4_21.
- [8] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469-483, 2009. doi: 10.1016/j.robot.2008.10.024.
- [9] Z. Zhu and H. Hu, "Robot Learning from Demonstration in Robotic Assembly: A Survey," *Robotics*, vol. 7, no. 2, Art. no. 17, 2018. doi: 10.3390/robotics7020017.
- [10] H. B. Suay, R. Toris, and S. Chernova, "A Practical Comparison of Three Robot Learning from Demonstration Algorithm," *Int J of Soc Robotics*, vol. 4, pp. 319–330, 2012. doi: 10.1007/s12369-012-0158-7.
- [11] S. Blankemeyer, R. Wiemann, L. Posniak, C. Pregizer, and A. Raatz, "Intuitive Robot Programming Using Augmented Reality," *Procedia CIRP*, vol. 76, pp. 155-160, 2018. doi: 10.1016/j.procir.2018.02.028.
- [12] J. Wu, D. Zhang, J. Liu and X. Han, "A Moment Approach to Positioning Accuracy Reliability Analysis for Industrial Robots," in *IEEE Trans. on Reliability*, vol. 69, no. 2, pp. 699-714, Jun. 2023.
- [13] "ROS," 2023. [Online]. Available: <https://www.ros.org/>. [Accessed: 12, May, 2023].
- [14] A. Mader and W. Eggink, 'The Creative Technology Design process'
- [15] C. ten Berge, "Reverse engineer discontinued KUKA I-do robots and make them applicable for educative use," 2022. [Online]. Available: <http://essay.utwente.nl/90481/>.
- [16] "KUKA iDo GitHub Repository," [Online]. Available: <https://github.com/g-mol/kuka-ido>.
- [17] "Codecademy," [Online]. Available: <https://www.codecademy.com/>.

- [18] R. A. El-laithy, J. Huang and M. Yeh, "Study on the use of Microsoft Kinect for robotics applications," Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium, Myrtle Beach, SC, USA, 2012, pp. 1280-1288, doi: 10.1109/PLANS.2012.6236985.