A logo with blue circles and black text

Description automatically generated

|  |
| --- |
| Technische Hochschule Ulm |
| Digital Twin of Kuka KR3 |
| Laboratory introduction |

|  |
| --- |
| Ibrahim Almohamed, Ahmed  04.12.2024 |

Contents

[1. Version and Control 2](#_Toc184250546)

[2. Glossary 3](#_Toc184250547)

[3. Prerequisites and Build Environment 5](#_Toc184250548)

[4. Templates 6](#_Toc184250549)

# Version and Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Name of Editor | Notes | Date |
| 1.0.0 | Ahmed Ibrahim Almohamed | n/a | 05.12.2024 |
|  |  |  |  |
|  |  |  |  |

# Glossary

|  |  |
| --- | --- |
| Term | Description |
| DT | Digital Twin |
| KukaDigitalTwin | A Digital twin system of the Kuka KR3 using ROS and Gazebo (simulation tool) . |
| AKL | “Automatisches Kleinteilelager” (DE) or “Automated small parts warehouse” (EN) |
| ROS | Robot Operating System |
| Kuka KR3 |  |
| KVP | KUKAVARPROXY |
| OPC-UA |  |
| SoftRealTime | system where deadlines are important but missing them occasionally does not result in system failure.(average delay of 5ms-30ms) |
| BiDirectionConnection | A connection between the physical and digital robots where commands can be sent from either robot to control the other, and the state information (such as position, velocity, sensor data, etc.) is continuously exchanged. |
| MoveIt2 | A robotic manipulation platform for ROS 2 and incorporates the latest advances in motion planning, manipulation, 3D perception, kinematics, control, and navigation |
| RosInterface | A software interface for the Ros2 to connect the Controllers and the simulation of Gazebo with the KVP protocol. |
| GUI | Graphical User Interface |
| RoboticsLab | A Laboratory at the THU that is used for running experiments of robotics. |
| KukaDigitalTwinDashboard | A Dashboard which is a part of the KukadigitalTwin GUI , used for control and monitor the digital twin and the real twin. |
| RosTasks | A RosTask is a software that aims to create a simple or complicated task for the KukaDigitalTwin , where the user writes a RosNode ,that is runnable on both the physical and digital twins. |
| RQT | RQT is a graphical user interface (GUI) tool for ROS 2. Everything done in RQT can be done on the command line, but RQT provides a more user-friendly way to manipulate ROS 2 elements. |
| RosNode | A node is a participant in the ROS 2 graph, which uses a client library to communicate with other nodes. Nodes can communicate with other nodes within the same process, in a different process, or on a different machine. Nodes are typically the unit of computation in a ROS graph; each node should do one logical thing. |

# Prerequisites and Build Environment

To successfully run the system, the following dependencies must be installed on your operating system:

1. **Docker**: Required to containerize and manage the application's services.
   * [Docker Desktop Installation Guide](https://www.docker.com/products/docker-desktop/)
2. **MongoDB**: Used as the database for storing application data.
   * [MongoDB Installation Guide](https://www.mongodb.com/docs/manual/installation/)
3. **Git**: Required for version control and to clone the application's source code repository.
   * [Git Installation Guide](https://git-scm.com/book/en/v2/Getting-Started-Installing-Git)

The System is running on a Docker Container that is built from Docker Image that is defined in a Docker file inside the Project Repository.

## Clone and Build

1. First Clone the Project from [GitHub](https://github.com/aialmohamed/MPA_KR3_Digital_Twin) .
2. Under "Path\To\Repository\MPA\_KR3\_Digital\_Twin\Software\Dashboard\DigitalTwin\bin\Debug\net8.0\DigitalTwin.exe" and run the exe file.
3. To use the Real Robot , first connect the Ethernet cable from your PC or Laptop to the socket like in the image.

Figure 1:Tcp Socket to the Robot

1. Then go to Windows Control Panel -> Network and Internet -> Change adapter settings -> Right-click on your Ethernet card -> Properties -> IPv4 and select Properties , change the ip address to something near the one on the robot (by me on the robot it was 172.31.1.197) so i set the ip on windows to : 172.31.1.100. click ok ( see image )

A screenshot of a computer

Description automatically generated

Figure 2: Set up Ip on your PC

1. Set The Robot on AUT Mode ( don’t forget to close all the windows of the Robot cabinet .
2. Run The Ros2\_gripper\_program and start the program but clicking on the Play icon until the robot is not moving anymore.
3. The Robot is then ready to work .

# Running The System

1. Start the GUI App (exe file)
2. Click on Register icon and Register with a username and password ( see image ) A screenshot of a computer

   Description automatically generated

Figure 3: Register Screen

1. After that click on Login icon (see Image) A screenshot of a computer

   Description automatically generated
2. After Logging in your Username shall be displayed under the KukaVerse Logo and text . A screenshot of a computer

   Description automatically generated

Figure 4: Username Display

1. Go to Launch to build the Image and Container ( if the image is not built then click on Build Image and wait until image is built ). A screenshot of a computer

   Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. If the image exists or its done building then click on Start container to start the system inside the docker container and wait the Opcua Server to come alive.

A screenshot of a computer

Description automatically generated

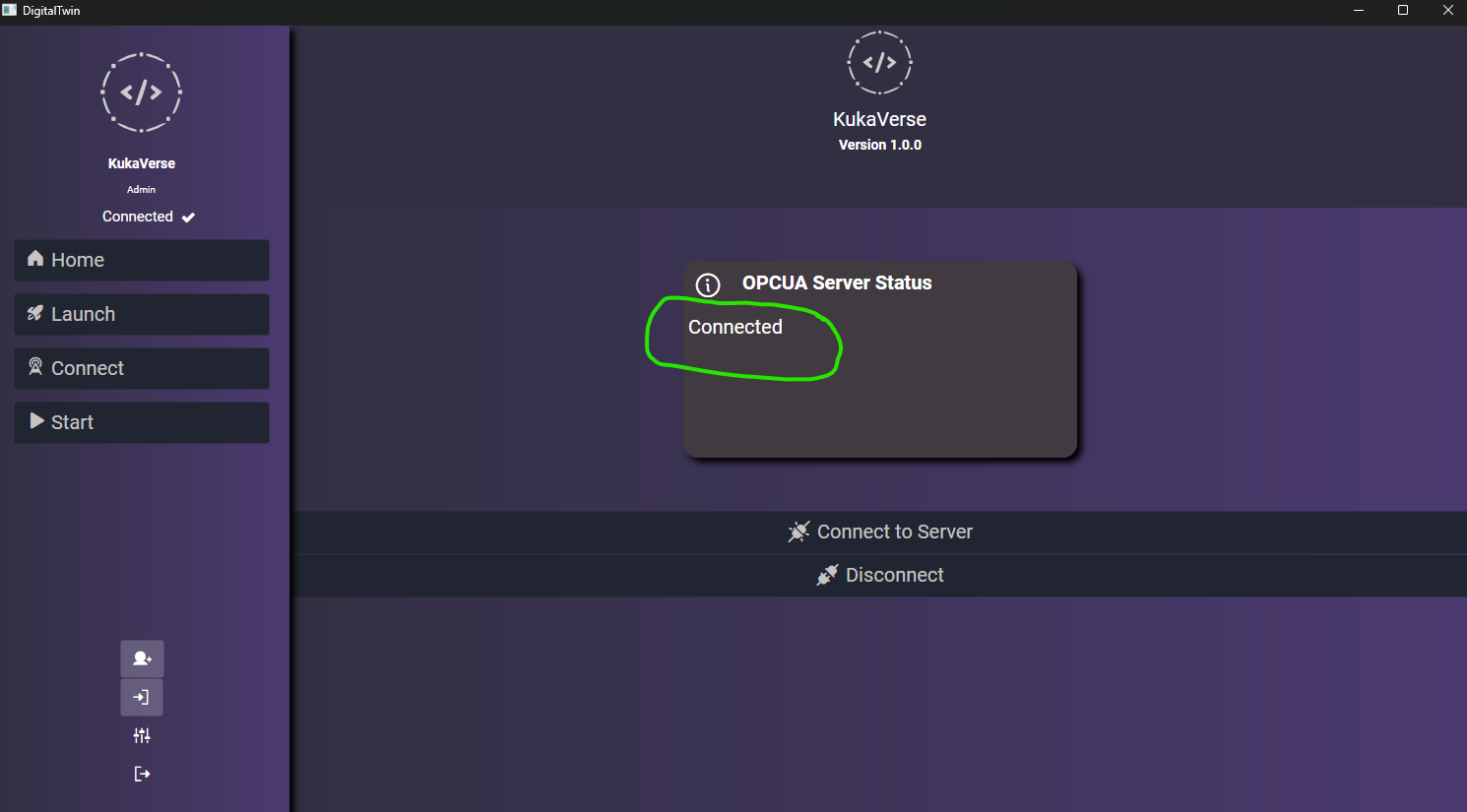
A screenshot of a computer

Description automatically generated

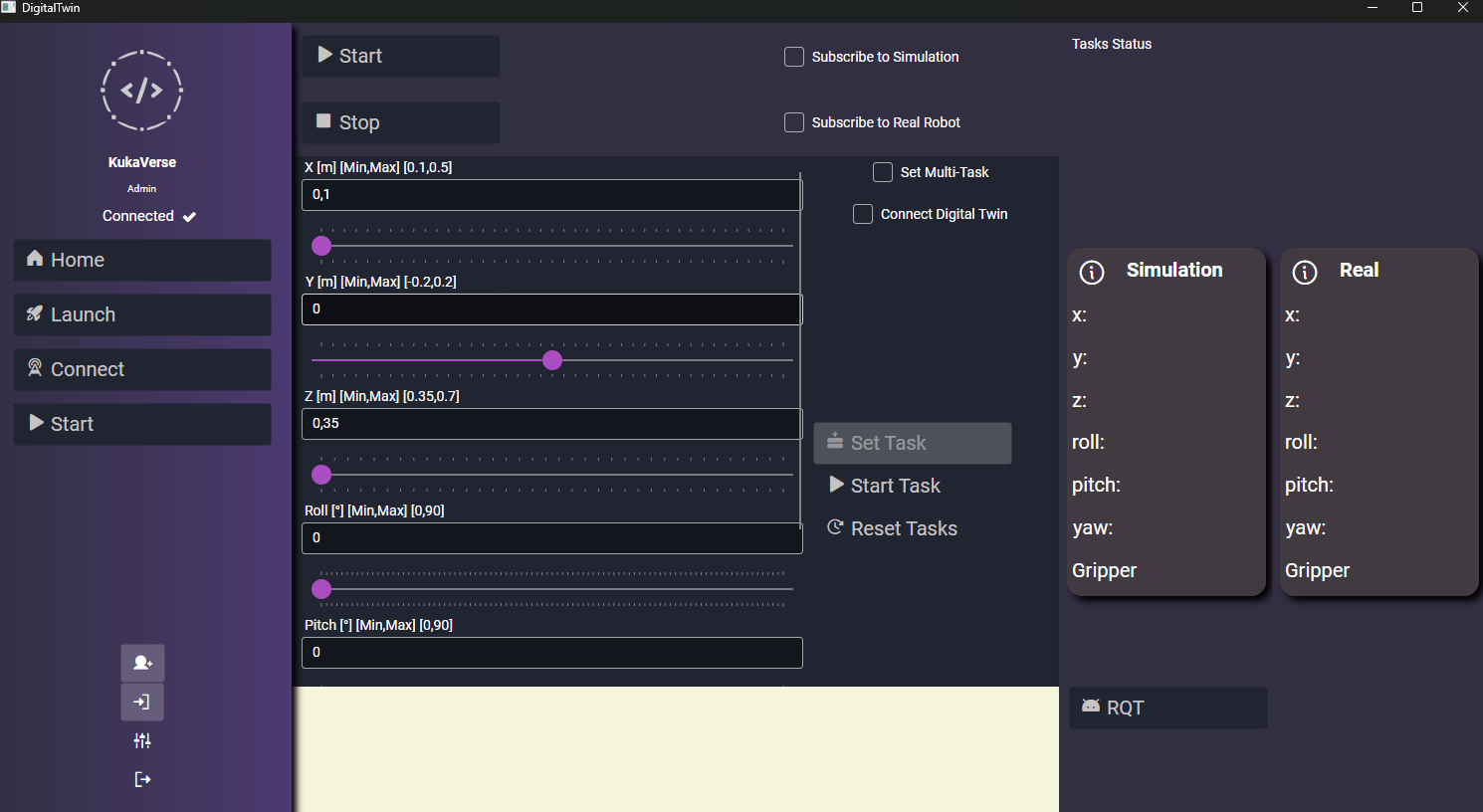
1. Go to Connect and click on “Connect to Server”

A screenshot of a computer

Description automatically generated



1. Go to Start to Start the System



# Templates

Requirements table

|  |  |
| --- | --- |
| Requirement ID |  |
|  |  |