A logo with blue circles and black text

Description automatically generated

|  |
| --- |
| Technische Hochschule Ulm |
| Digital Twin of Kuka KR3 |
| System Requirements |

|  |
| --- |
| Ibrahim Almohamed, Ahmed  21.08.2024 |

Contents

[1. Version and Control 2](#_Toc175199413)

[2. Glossary 3](#_Toc175199414)

[3. System Overview 5](#_Toc175199415)

[4. App Sketching 6](#_Toc175199416)

[5. System Use cases 7](#_Toc175199417)

[6. System Requirements 8](#_Toc175199418)

[a. Functional Requirements 8](#_Toc175199419)

[7. Templates 9](#_Toc175199420)

# Version and Control

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Name of Editor | Notes | Date |
| 1.0.0 | Ahmed Ibrahim Almohamed | Related to CustomerRequirementV1\_0\_0.docx | 21.08.2024 |
|  |  |  |  |
|  |  |  |  |

# Glossary

|  |  |
| --- | --- |
| Term | Description |
| DT | Digital Twin |
| KukaDigitalTwin | A Digital twin system of the Kuka KR3 using ROS and Gazebo (simulation tool) . |
| DashBoardOPCUA | A Dashboard subsystem that creates a OPCUA server . |
| ROS2KR3Core | A software based on ROS2 that runs and manages the simulation of the KR3 Digital twin. |
| ROS2KR3Connection | A connection method to connect the ROS2 (or the host PC) with the KUKA KR3 robot using KUKAVARPROXY . |
| ROS2OPCUABridge | a Software bridge to map a ROS2 node to a OPCUA node. |
| AKL | “Automatisches Kleinteilelager” (DE) or “Automated small parts warehouse” (EN) |
| ROS | Robot Operating System |
| Kuka KR3 | A robotic arm with a microscope  Description automatically generated |
| 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. |

# System Overview

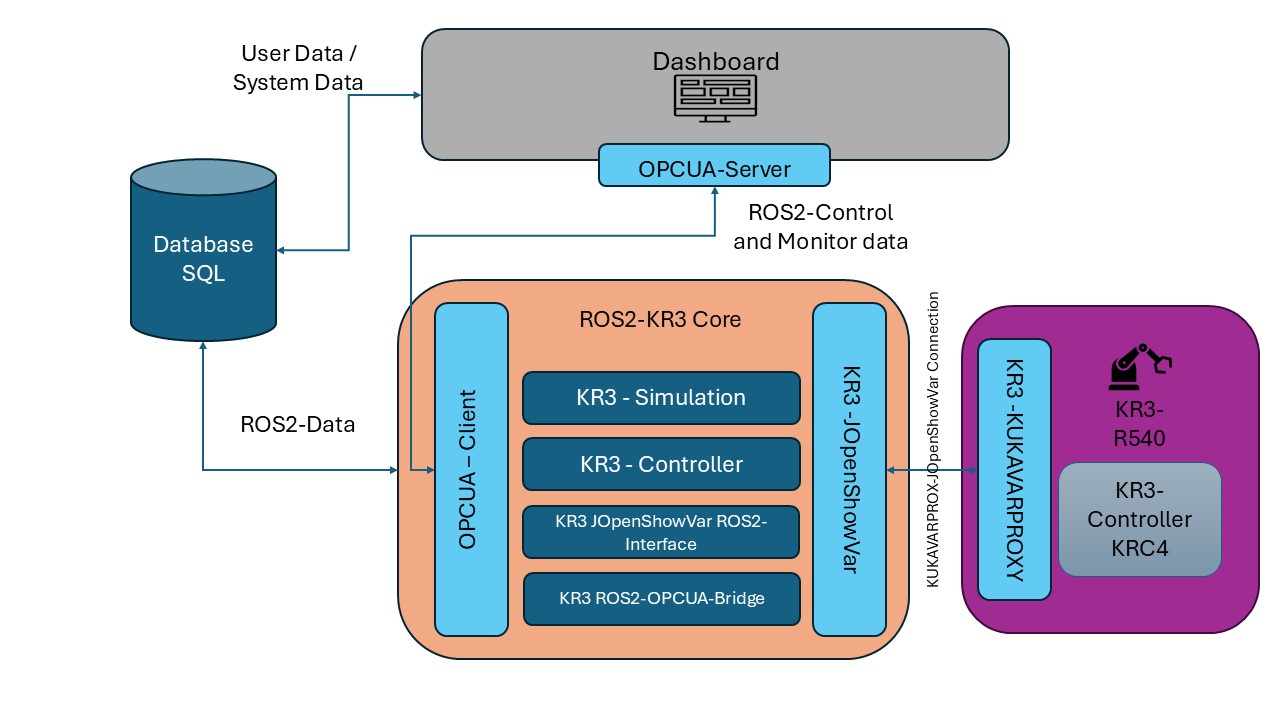


Figure 1: KR3 Digital Twin System Overview

The KukaDigitalTwin is a modular system that builds a Digital Twin of the KUKA KR3 using BiDirectionConnection.

The architecture is designed to scale, allowing additional robots or components to be added without major redesigns. The use of standards like OPC UA and the modular nature of ROS2 support this flexibility.

The KukaDigitalTwin is created from three main subsystems ,the DashBoardOPCUA , The ROS2KR3Core and the KR3-R540 physical robot system.

The KukaDigitalTwin creates also connection between the subsystems ,such as the ROS2KR3Connection and the ROS2OPCUABridge .

Also the KukaDigitalTwin contains a backend using SQL to store the user data , the System Data (from the Dashboard) and the ROS2 data .

In this Document it will defined the System use cases and the system requirements ,due to the System-Engineering approach of developing this system .

# App Sketching

# System Use cases

# System Requirements

## Functional Requirements

# Templates

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