SUMMARY

ROS systems are becoming more popular every year as part of the driving force of the Industry 4.0 Revolution due to their flexibility and ability to communicate. While standard ROS systems can only communicate locally, the use of MQTT as a middle layer opens new horizons for the ROS systems. Moreover, one of the solutions that can be seen on that new horizon is the remote Unity3D-ROS interface that was the original goal of this thesis and was achieved successfully.

The developed solution consists of a software package that can be integrated into Unity3D projects and a configuration approach that has to be used on the ROS systems to allow secure communication between the two. This solution also contains an additional Local ROS-Unity interface that uses ROS Bridge for local ROS-Unity3D interfacing.

The created solution was successfully integrated into the existing Digital Twing of the IVAR Laboratory with minimum alterations to the original project, thus achieving its goal of having the smallest possible code footprint and being easily integratable API.

The use cases of this solution – Boxbot prototype, Robotnik-RB2, and their Digital Twins – were successfully implemented and then presented in front of students, robotics specialist, and even the University's rector, Tiit Land. The solution performed admirably and without causing interference that could have disrupted other modules of the system.

The project, however, can be improved and expanded. Currently, there is no automatic transition between the Remote Mode, Local Mode, and ROS Bridge Mode, depending on the location and network configuration. This feature could greatly improve connection quality and vastly increase the reliability of the system.

To conclude this work, it should be said that remote communication for ROS systems could be an extremely valuable and practically viable feature that will have a great impact on industrial systems in the nearest future.