Open-Digital-Industrial and Networking pilot lines using modular components for scalable production

Grant Agreement No : **101017141**
Project Acronym : **ODIN**
Project Start Date : **1st January, 2021**
Consortium : **UNIVERSITY OF PATRAS – LABORATORY FOR MANUFACTURING SYSTEMS AND AUTOMATION**
FUNDACION TECNALIA RESEARCH & INNOVATION
KUNGSLIGA TEKNISKA HOEGSKOLAN
TAMPEREEN KORKEAKOULUSAATIO SR
COMAU SPA
PILZ INDUSTRIEELEKTRONIK S.L
ROBOCEPTION GMBH
VISUAL COMPONENTS OY
INTRASOFT INTERNATIONAL SA
GRUPO S21SEC GESTIÓN, S.A.
FUNDACION AIC AUTOMOTIVE INTELLIGENCE CENTER FUNDAZIOA
DGH ROBOTICA, AUTOMATIZACION Y MANTENIMIENTO INDUSTRIAL SA
PSA AUTOMOBILES S.A.
AEROTECNIC COMPOSITES SL. U.
WHIRLPOOL EMEA SPA
WHIRLPOOL MANAGEMENT EMEA SRL

Title : ODIN Digital Component – Final Version
Reference : D3.3
Availability : Public
Date : 17/01/2024
Author/s : LMS, KTH, TAU, VIS, INTRA, PILZ

Summary:
The purpose of this document is to describe the final version of ODIN Digital Component developments.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. EXECUTIVE SUMMARY

The main focus of this document is the presentation of ODIN Work Package 3 (WP3) modules' final prototypes up to M36 of the project. These components' final development is based on the ODIN Digital Component prototypes as described in D3.1.

The Digital Component is seamlessly integrated within the ODIN architecture, composed of the Open Component (OC), the Industrial Component (IC), and integrated all of them through the Networked Component (NC).

This deliverable contains the final deployment of the modules which composes the Digital Component, namely the Digital Resource Description (RD), the Digital Twin (DT), the Digital Simulation and the AI Task planner.

The Digital Resource Description Module, developed in task 3.1 is detailed in section 4 of this deliverable. The Digital Resource Description compiles the information about resources such as size, interfaces, geometry/kinematics, business and design properties, and the resources' specific capabilities. The different ODIN modules will have access to Digital Resource Description through the NC.

The digital twin, developed in task 3.2 and presented in section 5 of this deliverable. This module deploys an operating system of the digital twin to connect the virtual and real HRC cells.

The Digital Simulation, developed in task 3.3 and described in section 6 of this deliverable. Within this task have been developed the simulation components to create the virtual instance of the pilots and the digital twin in the virtual environment provided within Visual Components 4.0. The development of the virtual components is detailed in the deliverable and its integration with other ODIN modules.

The AI Task Planner has been developed in task 3.4 and presented in section 7 of this deliverable. This AI-based decision-making module provides task planning and dynamic reconfiguration capabilities to adapt the pilot to the changing production requirements with maximum efficiency.

2. INTRODUCTION

The availability of digital tools for design, configuration, validation, commissioning, and operation of robotics systems has been growing continuously during the last years. Despite the efforts to design and develop these tools, their potential has been limited for their utilization as standalone components, with limited integration with other tools and failing to demonstrate their potential and efficiency at large scale.

ODIN has overcome these limitations, developing the Digital Component, which provides a virtual instance of the pilot, implementing an accurate Digital Twin representation, able to interact with the physical system, that allows the commissioning, validation, and control of the pilot. The Digital Component (DC) targets to:

- Develop the digital resource descripting on the ODIN Open Component (OC) including the required information to digitally represent the OC.
- Build the digital twin using sensor data fusion to be used in the Human-Robot Cooperation (HRC) systems towards safe operation.
- Deploy the digital simulation environment for designing and validation of the pilot.
- Develop a decision-making AI module able to provide task planning and dynamic reconfiguration capabilities to adapt the pilot to the changing production requirements with the maximum efficiency.
- Integrate a digital simulation environment to deploy the virtual control and commissioning of the pilot line.

To achieve these objectives the ODIN Digital Component (DC) has been deployed in four different modules (Figure 1), developed in different tasks that provides the expected functionalities:

- Digital Resource description, task 3.1 and detailed in section 4 of this deliverable.
- Digital Twin, task 3.2 and detailed in section 5 of this deliverable.
- Digital Simulation, task 3.3 and detailed in section 6 of this deliverable.
- AI Task Planner, task 3.4 and detailed in section 7 of this deliverable.

The integration of the digital simulation environment to deploy the virtual control and commissioning of the pilots have been covered in D3.4 Digital Component virtual commissioning framework (final version) and D5.4 ODIN Industrial Component (refined version) and it's expected to be extended in D5.5 ODIN Industrial Component (industrial validation).

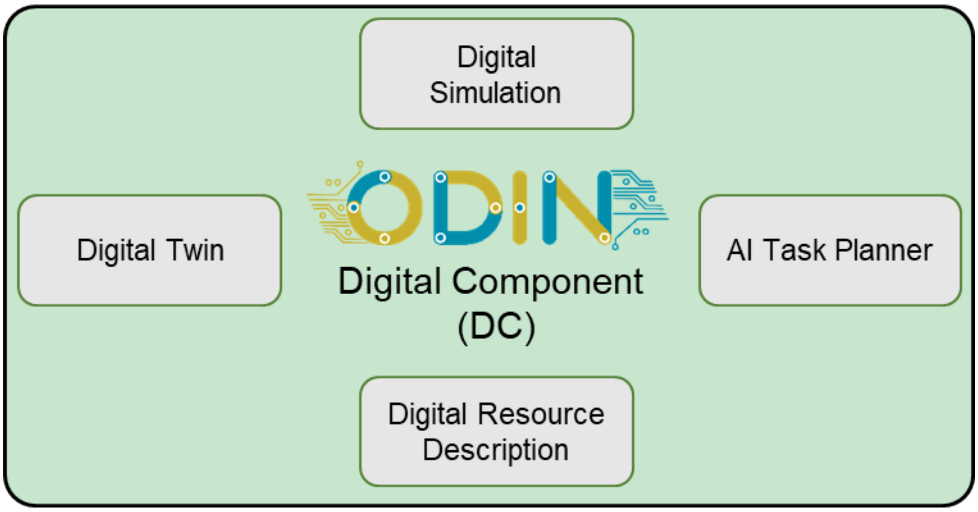

Figure 1: ODIN Digital Component (DC) modules

3. Digital Component (DC) Overview

3.1. Overall Digital Component architecture

The Digital Component (DC) under the ODIN perspective provides a virtual instance of the pilot implementing an accurate Digital Twin representation that allows the design, validation, commissioning, and control of the actual system (pilot). The DC integrates seamlessly into ODIN architecture through the Networked Component (NC).

Figure 2 shows the overview of the ODIN architecture, presenting the data flows between the Digital Component (DC), the Open Component (OC), the Industrial Component (IC) through the Networked Component (NC). Furthermore, the DC modules and the data flows are detailed.

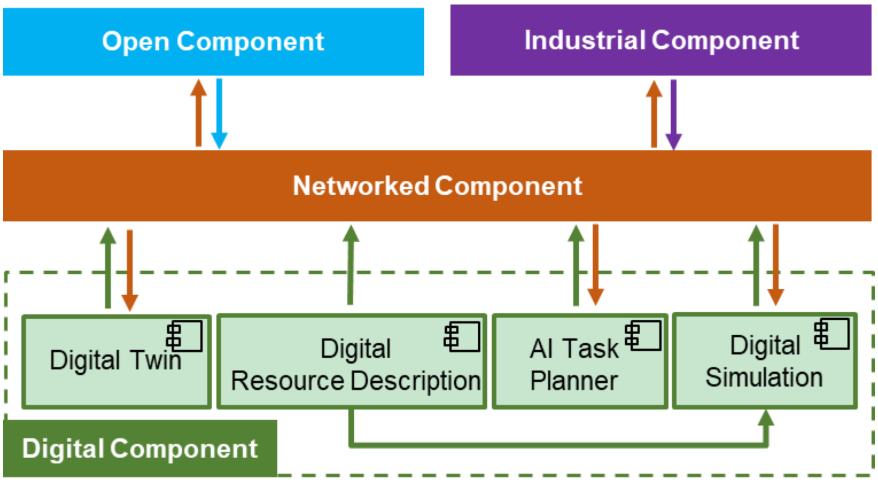

Figure 2: Overview of the ODIN architecture detailing the Digital Component and DC's modules integration

3.1.1. Digital Component (DC) – Networked Component (NC) connection

The NC consists of an integration architecture with open interfaces allowing the safe and secure communication of all robotics hardware and controlling systems, details of its final integration are provided in D4.4. As showed in Figure 2 the DC is directly connected with the NC providing different kind of data (resource models description, assembly task plans and task assignments, robot motion commands, etc.) but also receiving information about digital simulation data, alternative task plans validation, etc.

3.1.2. Digital Component (DC) – Open Component (OC) connection

ODIN OC targets the development, testing and integration of core robotics technologies such as mobile manipulators, reconfigurable tooling, perception systems and human interfaces using an open approach before their deployment in industry. D2.6 provides the details of the final integration within ODIN. The digital simulation model is used within ODIN for the virtual planning and validation of the pilots mirroring the OC in the virtual space. The OC also includes the Human Detection and Intention Estimation module which will provide data to ODIN Digital Twin though the OpenFlow module of NC.

3.1.3. Digital Component (DC) – Industrial Component (IC) connection

The IC focuses on the validation of the integrated solution at full scale under the actual production environments. It is responsible to provide robust data flow to the other ODIN Components for reconfiguration and optimization reasons. Even though the DC and IC of ODIN will not be directly connected, information derived from the DC will be transferred to the IC through the NC. The digital simulation module of the DC will also be related to the IC through the virtual commissioning framework which is described in D3.4 and the existing integration within the pilots in D5.4, which is expected to be extended in D5.5.

3.2. Main Features

This deliverable will present the final deployment of the ODIN DC modules:

- Digital resource description, (section 4)
- Digital Twin, (section 5)
- Digital simulation, (section 6)
- AI Task Planner, (section 7)

The integration of the modules within the ODIN architecture is done through the ODIN NC, which integrates OpenFlow. Figure 2 depicts the data flows between the DC modules and with other components, as presented in the section 3.1. The ODIN DC modules provide the following features,

- **Digital Resource Description**. The digital resource description includes all relevant features of the ODIN resources such as size, interfaces, geometry/kinematics, business, and design properties, in addition to representing the resource specific capabilities. This information is provided to the NC by the Digital resource description module. Information from this module to the digital simulation module is also interfaced though a dedicated interface visualizing the resources' geometry and their basic information.
- **Digital Twin (DT)**. The digital twin module deploys an operating system of the digital twin to connect virtual and real HRC cells. This module is a digital twin of ODIN pilots including sensor data for safe HRC. It is connected with the OC sending motion commands to ODIN robots. Additionally, the DT provides data to the NC about the position of the operators inside the industrial shopfloor. Real time sensor data is provided to the DT by the OpenFlow (NC).
- **Digital Simulation**. The digital simulation module provides the virtual representation of the pilot within the virtual space, mirroring the resources, products, processes, tasks and production flows. The base of the digital simulation is the virtual component. The digital simulation, communicates through the NC (OpenFlow) with other DC modules, to get product information from the RD, visualize sensor data within the virtual space from the DT and initialize simulation layouts with static and dynamic resources' data and receive generated task plans from the AI task planner.
- **AI Task Planner**. The AI Task Planner generates efficient tasks' planning and assignment to the available resources. Data transfer between the AI task planner module and the NC focus on the generated task plans provision to the OpenFlow and the receival of these task plans' evaluation data by the OpenFlow after their simulated execution by the Digital Simulation.

4. Digital Description of Open Components

4.1. Overview

Under Task 3.1, a formal comprehensive digital resource description for DCs is developed along with a service (Resource Catalogue Platform) for publishing this information for the other ODIN tools. All information targets to serve the easy and quick system design, reconfiguration, and integration of a production system - first the DCs in digital world and finally OCs as a functioning system. By using the digital resource model, each OC is described to form a digital resource representation (DC) in catalogue of OCs. The catalogue will be used as a source of OC/DC information for other ODIN tools.

The mentioned concepts are illustrated in Figure 3 with Resource Description (RD) and Resource Catalogue Platform (RCP). The former is a kind of formal electronic datasheet, representing comprehensive information about the production resource modules and the latter an online service for storing, distributing, and processing of those RDs. Integration to other ODIN tools, mainly the DCs and ICs, will happen through the interaction with the platform.

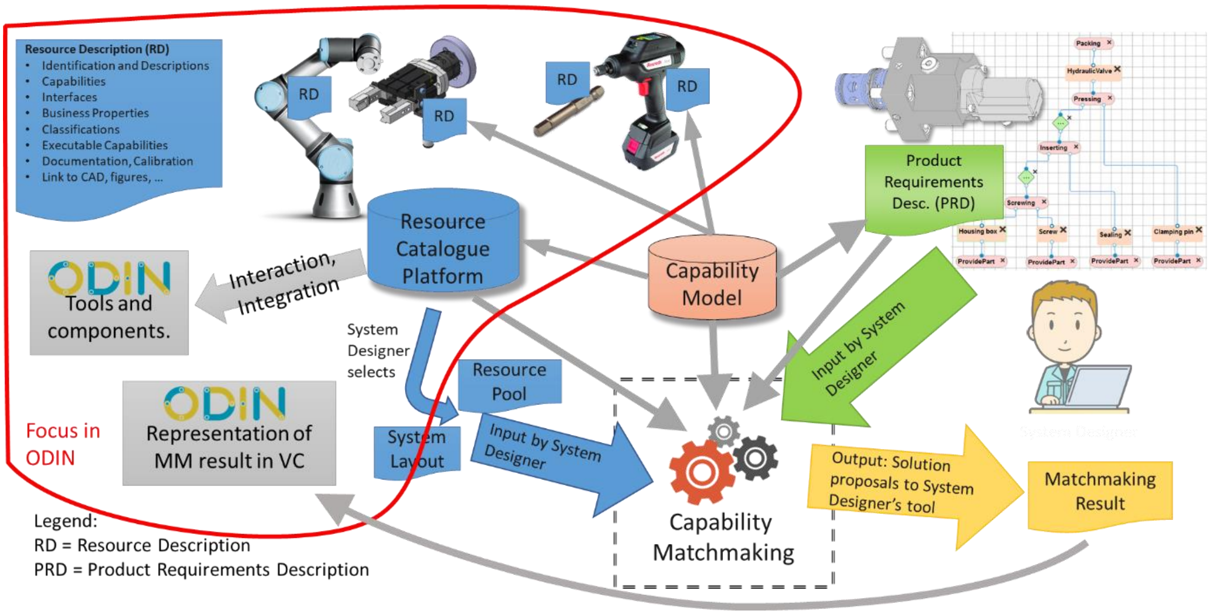

Figure 3: Overview of Resource Description and Resource Catalogue Platform concepts in ODIN

4.2. Overall methodology

The overall methodology for Resource Description includes 1) Resource Description data model; 2) RDs, which contain the content of production resources used in ODIN use cases; and 3) Resource Catalogue Platform (RCP).

4.2.1. Resource Description

4.2.1.1. Resource Description Concept

The Resource Description (RD) includes all relevant features such as size, interfaces, geometry/kinematics, business and design properties, in addition to representing the resource specific capabilities. A comprehensive mechatronic approach is followed, meaning that mechanical, electrical, control, and communication aspects are all thoroughly captured. Container model is followed, meaning that e.g. 3D models (CAD, URDF, VC, etc.), and datasheets and manuals provided by the vendor company are not replicated, but linked to the resource description with descriptive metadata. The main focus of resource descriptions is to comprehensively define the digital representation of production resources. This includes:

- Identification and Descriptions
- Classifications
- Capabilities/Skills
- Interfaces (mechanical, supply, electrical, communication)
- Business, technical and environmental properties
- Executable Capabilities i.e. abstraction of control interface
- Links to Documentation, Calibration
- Link to CAD, figures, etc.

The RD should provide the necessary information about production resources, which the other ODIN tools or e.g. system designers are needing (Figure 3). This information is however limiting to data related to catalogue resource, meaning the data which is shared among many instances of the same resource i.e. the product model or variant. Therefore, data model will exclude data associated to the instances of resources or their real time data.

The existing XML-based resource description concept and associated XML schemas, developed by TAU, will be utilized as a basis for the development. The resource description concept was initially developed during EU FP6 project EUPASS [1] in the form of initial idea. Concept was defined in PhD thesis of Niko Siltala [2]. During the EU Hz project ReCaM [3] it was developed on its next version, which included merge of two PhD thesis (Siltala and Järvenpää) together and further developments of the data model from the project needs. The data model has been assessed at the ODIN project and evaluated how the model corresponds the needs expressed here. The analysis and indications showed that only small additions are needed to the data model to serve ODIN better. These additions include adding foreign keys to link RD with other SW, which are utilising their own ID systems, such as Visual Components, and to create a mapping between RD's ID and other tools.

4.2.1.2. Resource Description Data Model

The data model of resource description is formalised and documented, and it is represented as XML Schema (XSD) [4]. This is the ODIN updated version of RD's XSD. The XSD of RD will define the names, quantity and relations between XML elements forming the data model. The attributes and their cardinality are also defined. In case of enumerated values, the enumerations are defined.

The capabilities follow ontology model in format of RDF/OWL. It is not specified in format of XSD, but content is injected inside a RD into specific XML element. The content and semantics of capability model is defined in its own OWL model, which is shared online [5]. In case new capabilities are needed, they need to be specified and added to the shared online model before they become active and used within the RDs.

4.2.2. Modelling the Resource Descriptions for ODIN

The resource modelling for the ODIN resources has been performed. The data about the resources used in ODIN is gathered from the other deliverables, specifically from D1.2, D2.2., D2.6, D5.1; WP presentations in various occasions; and bi-lateral communication with T3.1 and other partners. Naturally, the web pages, datasheets, manuals, and other documentation provided by the technology vendors of these production resources (robots, grippers, mobile platforms, etc.) have been used as data sources for RD modelling.

The modelling of RDs has been performed by T3.1 with help of RD Editor application developed earlier, and with generic XML editor, which can validate the RD against its XML Schema (XSD). The former is a partial implementation, but it is even though very helpful for modelling the capabilities. It connects directly to the Capability Ontology OWL model shared online, and provides a GUI for adding capabilities and their parameters on the RD. It also keeps the parameters and possible objects consistent with the Capability Model.

The list of production resources modelled in ODIN are listed in the section 4.4.1 / Table 1. Modelling of some resources is still on-going, and those will be completed during the last year of the project. Priority has been the TAU small scale pilots and White Goods use case, where TAU´s main contributions are also focused on.

4.2.3. Resource Catalogue Platform

This part of the task targets to create a platform for the catalogue to search and distribute the resource descriptions. The main identified use cases for Resource Catalogue Platform (RCP) are illustrated in Figure 4. The use cases are reflected through two main actors – *Resource user* and *Resource provider*. Examples of resource users are system designers, system re-configurators and system integrators. The RCP can be used as a service by another software, which accesses this service.

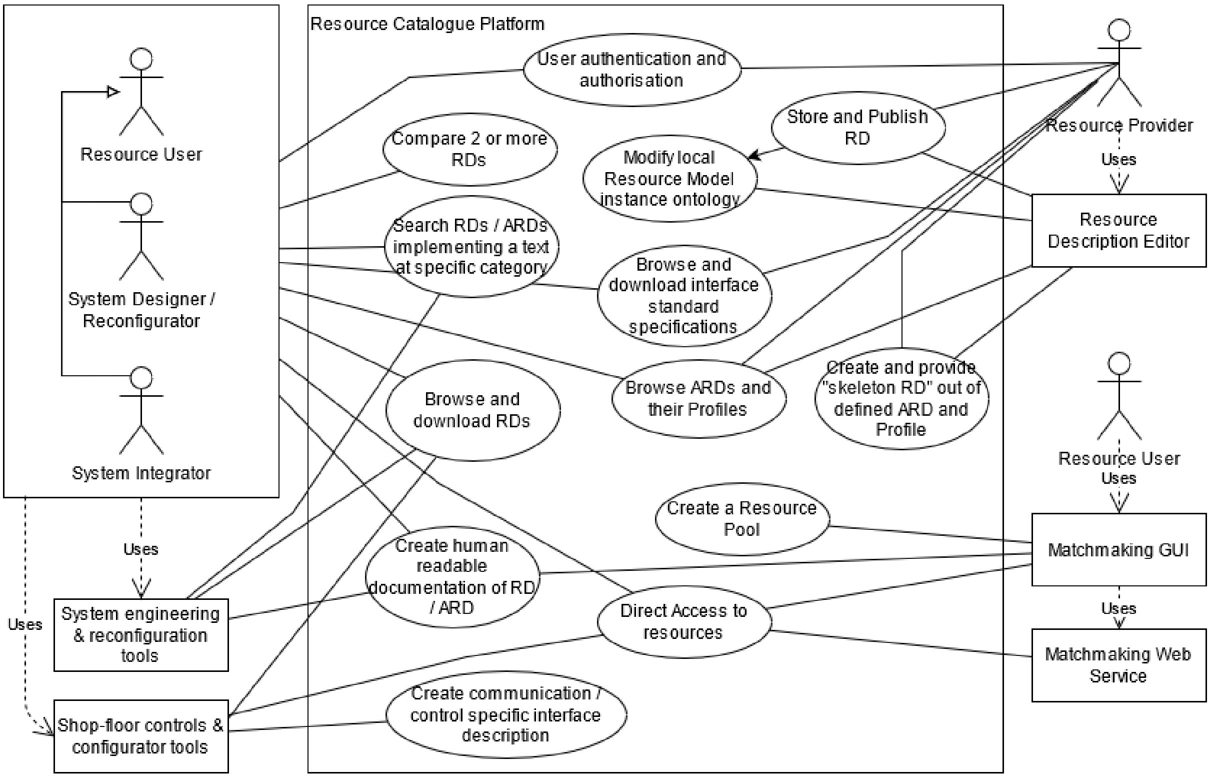

Figure 4: Use cases for Resource Catalogue Platform [6]

The main use case at RCP for a *Resource user* is *Browse and download RDs*. The previous can be appended with search (*Search of RDs / Abstract Resource Descriptions (ARDs) implementing a text at specific category*) and *Create human readable documentation of RD / ARD*. Interface standard information can be browsed and downloaded if needed, if those are not coming from international or company specifications, which are published already elsewhere.

The main use cases for the production *Resource Provider* are creation of RD and *Store and Publish RD*. The former can be supported by *Browse ARDs and their Profiles*; *Browse and download interface standard specifications*; and *Create and provide "skeleton RD" out of defined ARD and Profile*. These all are helping the resource provider to make correct selections with the RD definition e.g. for the selection of right interface specification.

In the early phase of the detailed design of new RCP, it was decided to advance in clear steps. The ARD concept was excluded from the project, to focus on making the RD section first implemented properly, because this is the key data for user and needed first when applying the Resource Description Concept. The architecture of the RCP was also carefully specified (Figure 5). It's reasoning and findings are discussed in [10]. The selected architecture follows clear distinction between back end and front-end applications. The back end is implemented as RESTful web service having direct connection to IBM DB2 database. DB2 was selected, because it can store both relational data but also XML data, which can be directly queried. The GUI for RCP is the front-end application implemented as React web application. User can use their favourite web browser to use this front-end application and access all the RD data. These both implementations are discussed more in details in the following sections 4.4.2 and 4.4.3 respectively. The other ODIN SW applications can integrate to RCP utilising the same RESTful API, which is used by the front end.

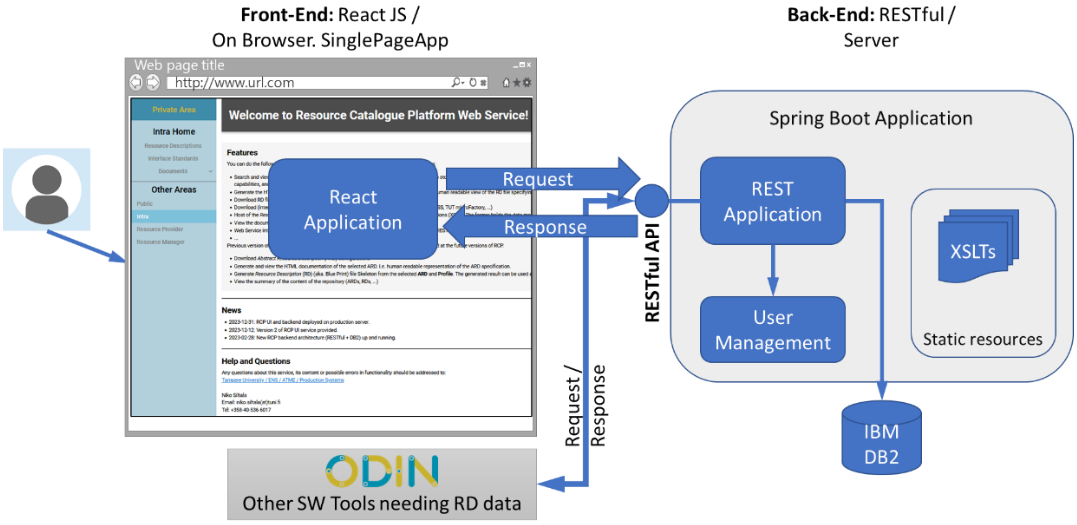

Figure 5: Architecture of Resource Catalogue Platform

4.3. Integration of Digital Descriptions with Other ODIN Tools

The Resource Descriptions will work as source of resource information for ODIN components (NC, OC, and IC) and modules. The primary intended usage of RD data is for the system design and reconfiguration phases. During the system operation the RD is not intended to be extensively utilised, but it can be used for the view and identification of the resource, or to access it via the manuals and other documentation of the resource. E.g. while an operator or maintenance staff is adjusting or maintaining the module or sorting out some problem.

The back end of resource catalogue platform provides a RESTful API for accessing RD information, which is an access point for other ODIN applications to this data as well. The API provides various endpoints to interact with the platform and the provided data. Some endpoints will provide specific data or view related to RD, but also the whole resource description data is provided as complete XML formatted RD. The implication for the receiver is that it needs to understand and parse the full RD data model. Should a more human friendly view to RD needed, there is an endpoint to get the RD data as HTML formatted document. The API is documented dynamically online with the help of Swagger [7]. Thus, the documentation remains all the time valid; it can be used to test the interactions with the published endpoints; and the API description can be exported and then imported to any other application for easier integration.

OpenFlow [8] (T4.1) is one of the main tools in ODIN integrating to the RCP. OpenFlow has very important role in ODIN production application generation, as it is the middleware and orchestrator application for ODIN components. The OpenFlow will utilise the data of RDs in various ways. The main use case is to visualise the resource (icon and name) on its UI and providing the user of OpenFlow a detailed view and information about the resource they are interested on. Other tools can access key RD resource data through the OpenFlow, which forwards the requests to RCP back end. The OpenFlow utilised limited set of the provided RESTful API, and the full API is only available through direct connection to RCP. Other ODIN tools connected to OpenFlow can access those data of RDs, which is made available by OpenFlow, and which it relays forward. Then other tools can always use the RCP API to connect to RCP and to directly retrieve all necessary information from the catalogue. Integration between RCP back end and OpenFlow is completed.

Through the OpenFlow integration the Resource Description data is made available to all ODIN pilot use cases. The UI of OpenFlow can visualise the RD data when the operator uses the OpenFlow application, or when the manufacturing sequence is executed by the OpenFlow. The OpenFlow can visualise the RD as icon, simple key data (Figure 9) or even through the full human readable content of the RD.

Integration to Visual Components 4.0: A prototype tool has been implemented for integration of TAU's matchmaking tool result (Figure 3 bottom right corner) and resources (RDs) listed on it to Visual Components 4.0 (VC 4.0). The tool is intended for production system design or reconfiguration phase, and it works as plugin in Visual Component's software. The tool gets a matchmaking result as input and after operator clicks the trigger button the process will start. First matchmaking result is processed through, then corresponding resource IDs in VC 4.0 are found out from mapping table, and finally resources are placed on canvas at VC according to the matchmaking results, grouped per each match. The additions to RD data model (foreign keys) are supporting on dynamic generation of the mapping table. Results of this development are reported in [8].

The foreign key concept and resource mapping can be utilized also in other scenarios. For example, when the AI Task Planner (T3.4) is creating new plans and simulates them with the help of Visual Components 4.0 (See section 7.2, Figure 67). The necessary mapping data between the applications can be dynamically generated from the content stored into the RDs.

Integration to Tecnalia's OISP (T2.4): Use case scenario for this is that a new resource is acquired, vendor provides a RD with it, and how the new resource can be easily introduced into the OISP (On-site Interactive Skill Programming) system. There is on-going research with Tecnalia to find out, can the data from executable capability section of a RD data model to be used to automatically generate a description of skill/action/service model which they use. The generated content can be then imported to Tecnalia's OISP system as a stub for a new resource, and then this resource stub can be finalised manually by the OISP expert user. Finalisation includes some choices and selections, which are not present in RD data model, and addition of application specific data.

4.4. Final Implementation

First the subsection represents the status of the resource modelling in the ODIN. Continuing with the implementation of Resource Catalogue Platform.

The requirements of Resource Catalogue Platform are defined in [6]. From these requirements architecture (Figure 5) and detailed specification were conducted. The specification and implementation itself are split into two halves – Back end 4.4.2 and front end 4.4.3 applications. These are reporting the final implementation of the RCP.

4.4.1. Resource Descriptions for ODIN

The list of production resources modelled in ODIN are listed in the Table 1. The location column defines the ODIN use case, where the resource is primarily used. *ODIN_All* means the resource is used in two or in all three ODIN use cases. *ODIN_WG* is the White Goods use case and *HRC Pilot* means the TAU's small scale pilot environment. Some resources from earlier list (D3.1) are excluded because they are not used in ODIN use cases. For some cases the resource modelling is still on-going, and details for those RDs will be updated. Full modelling of the whole list will be completed during the last year of the project since this is linked as an integrated activity with the use case demonstrations. So far, the priority has been the White Goods use case and TAU's small scale pilots, where TAU's main contributions are focused on.

From the Table 1 can be seen that totally 34 RDs are identified for ODIN use cases. Out of these resources 24 are modelled as RDs, and 10 (#17 - #19, #28 - #34) will be updated during the last year of project. The number of modelled resources can still increase if more resources are identified important for the ODIN use cases.

Table 1: Production resources to be modelled with RDs in ODIN

#	Location	Category	Manufacturer	Device	Model
1	ODIN_All	Camera	Microsoft	Azure Kinect	-
2	ODIN_All	Camera	Roboception	RC Visard 65	-
3	ODIN_All	Camera	Roboception	RC Visard 160	-
4	ODIN_WG	Fixture	LMS/Wingman	Tool changer holder	Wingman XXX
5	ODIN_WG	Gripper	LMS	LMS Vacuum gripper for WhiteGoods	-
6	ODIN_WG	Gripper	LMS/Festo	LMS Festo flexible gripper for WhiteGoods	FESTO DHEF-20-A
7	ODIN_WG	Gripper	LMS/Schunk	LMS Magnetic gripper for WhiteGoods	Schunk EMH-RP 036-B
8	ODIN_WG	Other	Optoma	Projector	Optoma EH412STX
9	ODIN_WG	Robot	UR	UR 10 collaborative	UR 10
10	HRC Pilot	Gripper	TAU/Schunk	TAU_Schunk_ThreeFinger gripper	PZN-plus-100-1
11	HRC Pilot	Gripper	TAU/Schunk	TAU_Schunk_TwoFinger gripper	PGN-plus-P-80-2
12	HRC Pilot	Robot	ABB	IRB 4600	IRB4600 40 255
13	HRC Pilot	Robot	ABB	IRB 4600	IRB4600 60 205
14	HRC Pilot	Robot	UR	UR 5	UR 5
15	ODIN_Aero	AGV	Robotnik	Robotnik mobile platform	RB-Kairos
16	ODIN_Aero	Robot	KUKA	KUKA LBR iiwa 7 800	LBR iiwa 7 R800 MF SC FLR
17	ODIN_Aero	ProcessTool	Setitec	SETITEC Automated Drilling Unit	Setitec ST1200
18	ODIN_Aero	Safety	PILZ	PILZ Safety Radar	-
19	ODIN_Aero	ToolBit	Spacematic	Drill for Aero	Hi-shear Spacematic M1000
20	ODIN_Auto	AGV	AIC	Mobile platform	-
21	ODIN_Auto	AGV	Comau	Agile 1500	VCRGR-0000117780
22	ODIN_Auto	ProcessTool	OnRobot	Screwdriver	OnRobot Screwdriver 103961
23	ODIN_Auto	Robot	ABB	IRB 4600	IRB4600 45 205
24	ODIN_Auto	Robot	Comau	Aura Cobot	AURA-170-2.8
25	ODIN_Auto	Robot	Comau	Racer5	Racer-5-0.80 COBOT
26	ODIN_Auto	ToolBit	OnRobot	Tool bits for screwing	
27	ODIN_Auto	Adapter	Tecnalia	Adapter: Schunck SWS - OnRobot	Tec Adapter SWS-OnRobot
28	ODIN_Auto	Conveyor	DGH	Conveyor - curve	-
29	ODIN_Auto	Conveyor	DGH	Conveyor - straight	-
30	ODIN_Auto	Gripper	LMS	Gripper for motor	-
31	ODIN_Auto	Gripper	LMS	Gripper for gearbox	-
32	ODIN_Auto	Pallet	Stellantis	Pallet for engine	-
33	ODIN_Auto	ProcessTool	ESTIC	ESTIC Cordless Handheld Nutrunner	ESTIC Cordless Nutrunner
34	ODIN_Auto	Safety	PILZ	PILZ Laser scanner	PSENscan

4.4.2. Resource Catalogue Platform – Back end

The back end application is providing a web service layer for RCP. It provides various services to view, search, and manipulate RD data. It should help on storing and distributing the RDs and make them available for downloading. The services must support for searching suitable RDs by name, categorizations, capabilities, and interface standards, which can be achieved by implementing a search functionality. Additionally, the software must provide simple processing operations for RDs, including generating human-readable documentation, creating RD data structure, and performing other transformations.

The back end application is a server solution implemented in Java on top of Spring Boot and Apache Tomcat. It is implementing a RESTful API for other applications to utilise these services and retrieve data from it. The API is documented dynamically online with the help of Swagger [7]. The IBM DB2 database is used to store the content, and the back end application offers an bidirectional access to this data. Implementation and reasoning behind the selections made are discussed and detailed in [10].

The service and access to various endpoints is role based. Five distinguish roles are defined – public, normal user, resource provider, resource manager and admin. The public page section is accessible to anyone without the need of login. The same applies to accessing specific RD or additional resources like data model (XSD) or transformations (XSLTs). One needs to know the exact URL of a resource to access these. Public pages are not allowing any search, browsing, or listing services, but it is important that a resource is retrievable without login if integrated application requires to show some key data.

Normal user is the first one which needs to log in to the system. This role is the main user role for RCP application. Most of the registered users would fall on this role. They can view, search, and retrieve RDs from the service. Also, application to application integration is expected to utilise this role. E.g. when OpenFlow service is integrated, it should have its own user account available. The Resource Provider is a role which vendors of production resources should have. The are the makers of new RDs representing their machine components. They have the same functions available as normal user, but on top of that they have possibility to upload new RDs to the system. They can add additional attached files to the service (such as images of the resource); update and validate the uploaded RD; and finally send it to review before publishing. The role of Resource Manager is to perform the review of RDs submitted by the Resource Providers. They are as well responsible for harmonizing the capability and interface standards information. They should check and verify appropriateness of content in submitted RDs and maintain the lists of interface standards and standardisation bodies. Once the content of submitted RD is verified to be in-line with the standards, the Resource Manager can accept the RD and it is getting published in the service. The final role is Administrator, who are responsible for admin the whole application. They have the right to add new users and companies to the service and to modify them. Also, they are the ones granting the different roles for the other users.

The back end application and its REST API has been implemented. The final ODIN M36 version includes 56 implemented endpoints. As total, the 74 endpoints are specified, but some of them are excluded from the project. Most of the excluded ones relates to concept of Abstract Resource Description (ARD), which was decided to be excluded from the project. The REST API is available from https://resourcedescription.rd.tuni.fi/rcp/swagger-ui/index.html . See Figure 6. Some of the endpoints, which are targeted for administrative purposes only, are not exposed to the publicly available list in the Swagger UI.

The features the endpoints of RCP back end are represented and discussed more in detail in the following section 4.4.3 where the front end of the RCP application is discussed.

4.4.3. Resource Catalogue Platform – Front end

The front end of the Resource Catalogue Platform (RCP) empowers users to seamlessly interact with its various features and functionalities. The design and planning of the front end is made with help of Figma online tool. All the views were designed and tested there before the implementation started. The front end is built on ReactJs, a powerful JavaScript library for developing dynamic and responsive user interfaces. It aims to deliver a user-friendly and efficient experience with the use of different technologies and thoughtful design. The usability of the RCP front end application is evaluated and assessed in [11]. The RCP front end GUI application is available in https://resourcedescription.rd.tuni.fi/rcp/ui/ . Screenshot of this interface is shown in Figure 6.

Complementing React, we utilize various technology stacks that include popular libraries such as react-select, react-table, react-toastify, and others as mentioned in Table 2. This combination of tools ensures a feature-rich and visually appealing front end, aligned with industry best practices.

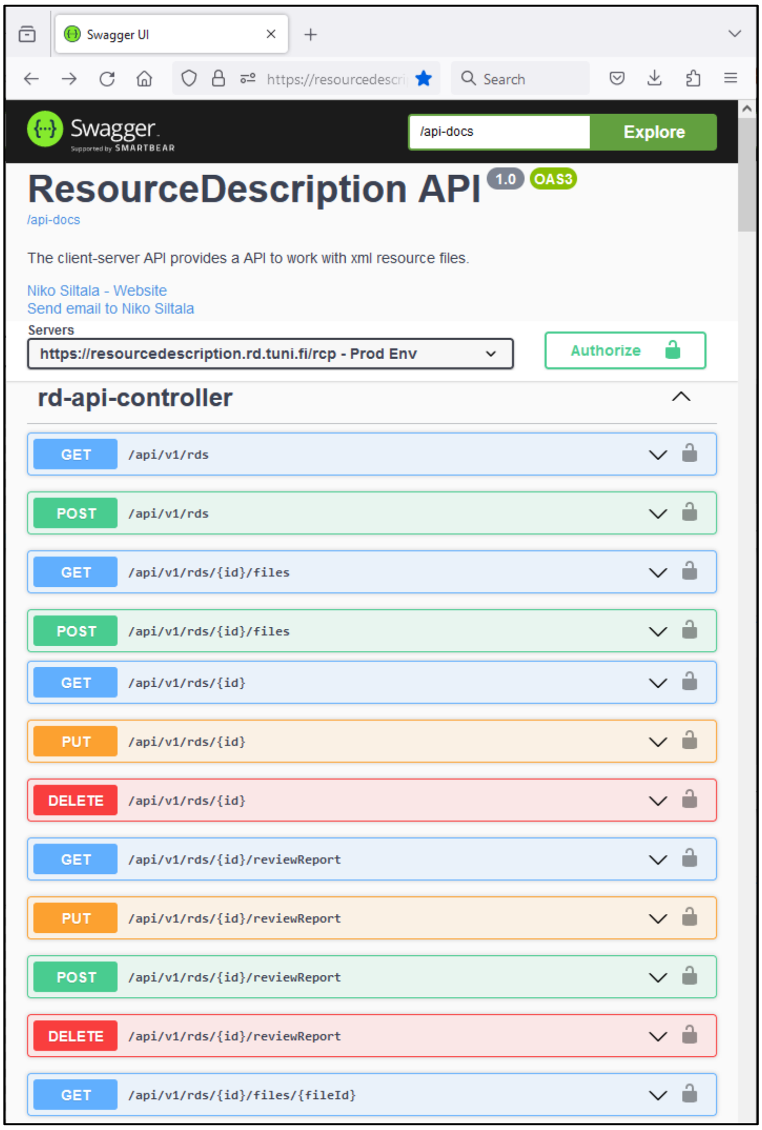

Figure 6: A few endpoints of RCP backend application represented in Swagger UI

Table 2: Library and Tools used in UI development of RCP

Library & Tools	**Descriptions**
React-Select	Library designed for creating flexible and customizable “Select Input Control” in react application
React-Table	Library designed for creating dynamic, extendable, and accessible tables in React.
React-Toastify	Provides a way to show notifications in the React application.

Library & Tools	**Descriptions**
React-Quill	Rich text editor for react application, allowing users to format text with several styles.
Redux Toolkit	Simplifies the process of managing state in react application.
Moment.js	Library used for parsing, validating, manipulating, and displaying dates and times in JavaScript.
Js-file-download	Library that simplifies the process of triggering file download.

As users interact with the Resource Catalogue Platform's front end, their experience is shaped by various roles available within the application. To get started, users are prompted with a login popup window (Figure 7), providing a secure gateway to their personalized functionalities. Therefore, the user of RCP must have a valid user account a prior.

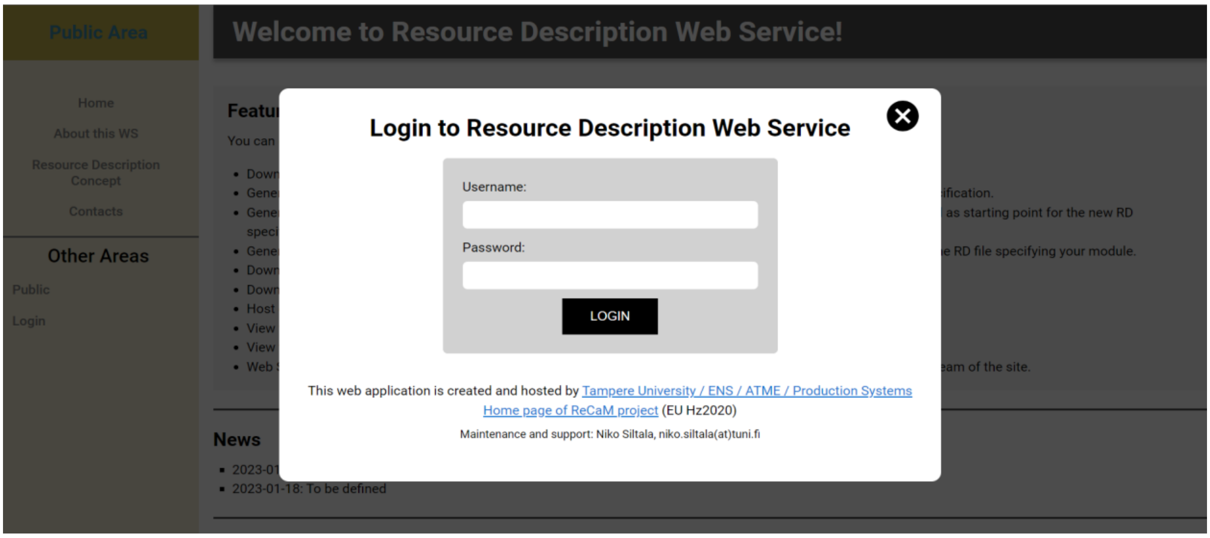

Figure 7: **User Interface of a login popup**

Upon logging in, users can explore a diverse range of features aligned with their assigned roles. The Resource Catalogue Platform (RCP) encompasses various roles, such as **User (Resource), Resource Provider, Resource Manager**, and **Admin**. Notably, features accessible to a user with a normal "user" role are also available to those with roles like resource provider, resource manager, and admin. However, the reverse is not applicable, ensuring a nuanced and role-specific responsibility and functionality. Every authenticated and authorized user can access and interact with the interface assigned to the "user" role. Registered user account can have multiple roles assigned simultaneously.

4.4.3.1. User (Resource)

The features available to RCP users in the "Normal User" role are outlined below, accompanied by their respective user interface screenshots.

4.4.3.1.1. View, Download, and Filter Resource Description

Resource Users, as the primary users of the Resource Catalogue Platform, benefit from a seamless experience in viewing, filtering, and downloading Resource Description (RD) files. Figure 8 illustrates clearly visible icons that allow users to effortlessly perform these actions per corresponding RD. The view action represents Resource Description files in a human-readable HTML format for easy comprehension. Moreover, users can download the Resource Description in its native XML format, with the corresponding action button. Notably, users have the capability to filter the table based on RD's internal attributes, such as Capabilities and Interface Standards, as demonstrated in Figure 8. For a quick preview of key data of selected RD, users can simply click on the first column of each table data row, as depicted in Figure 9.

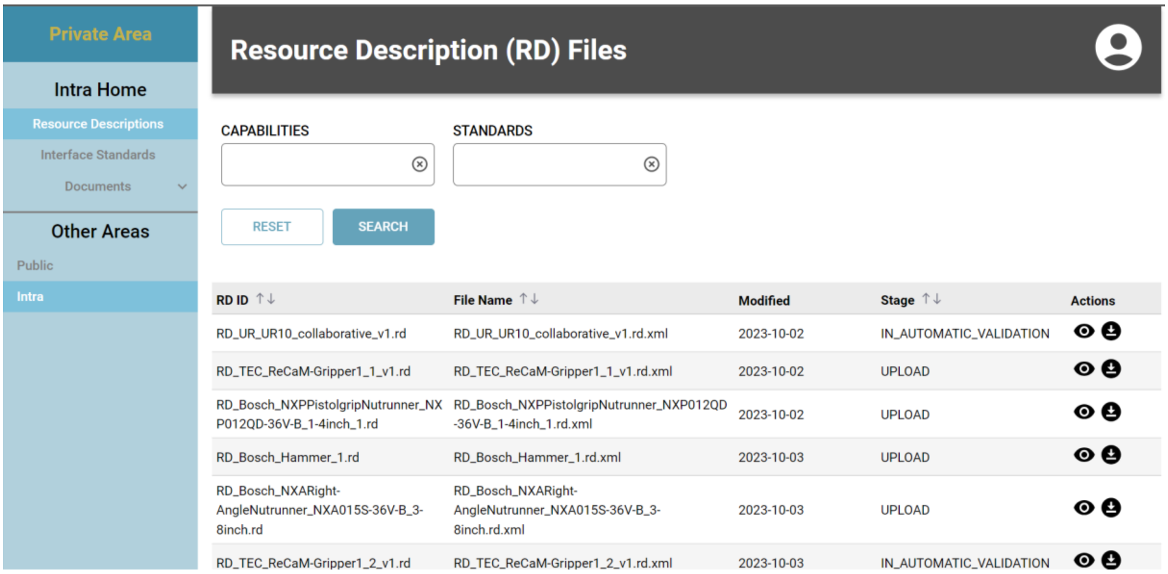

Figure 8: User Interface of Resource Description page

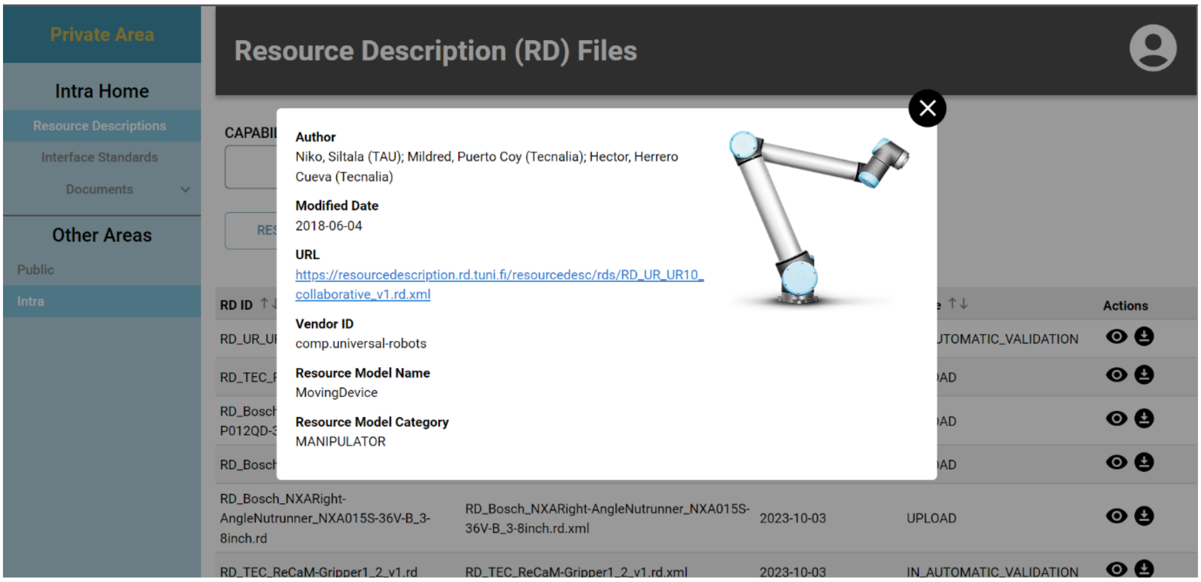

Figure 9: Quick view mode for single Resource Description item

4.4.3.1.2. Filter Interface Standard Used in Resource Descriptions

The interface standards are representing and documenting the interfaces used to connect production resources together. Users can apply targeted filters, enhancing their search experience to focus on the list of Interface Standards used in Resource Descriptions. These filters can be done based on various attributes such as Code, Name, and Description as illustrated in Figure 10.

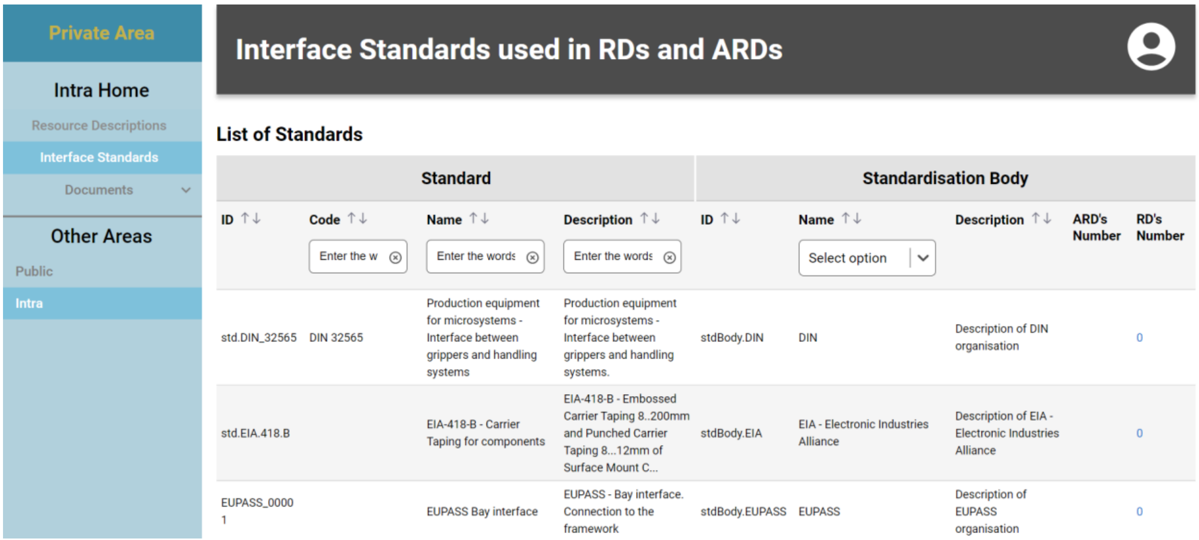

Figure 10: **User Interface of Interface Standards used in RD**

4.4.3.1.3. View Resource Description list implementing a specific Standard

Users can seamlessly navigate through the list of Resource Descriptions that have implemented specific Standards. By clicking on the "**RD's Number**" titled column at the end (Figure 10), users can explore the RDs implementing a dedicated interface standard and accessing related standard data. This action leads to an interface, as illustrated in Figure 11, providing users with a list of Resource Descriptions associated with selected Standard.

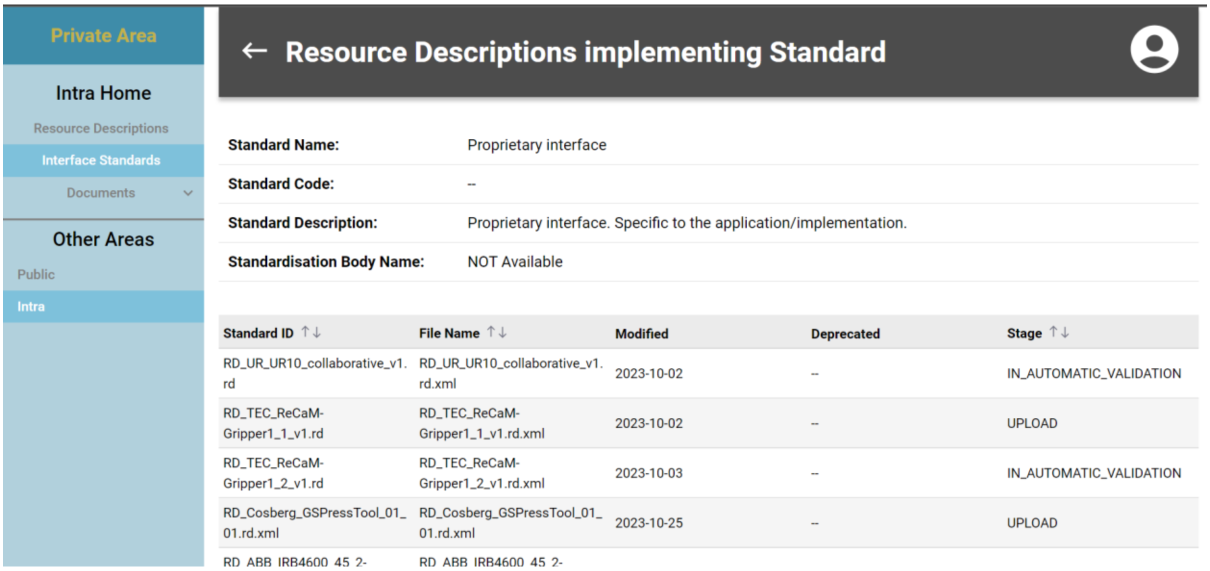

Figure 11: User Interface of RDs list implementing Standard

4.4.3.1.4. View Standard Specification Documents, XML Stylesheets, and XML Schemas

Users have the option to engage with an UI viewing a comprehensive list of files related to the Standard Specifications, XML Transformations (XLSTs), and XML Schemas (XSDs). These are stored and distributed via the RCP service for the public and for other applications to use. E.g. other applications can use the XSDs for validating the RD data model. This functionality is showcased in Figure 12, Figure 13 and Figure 14 respectively.

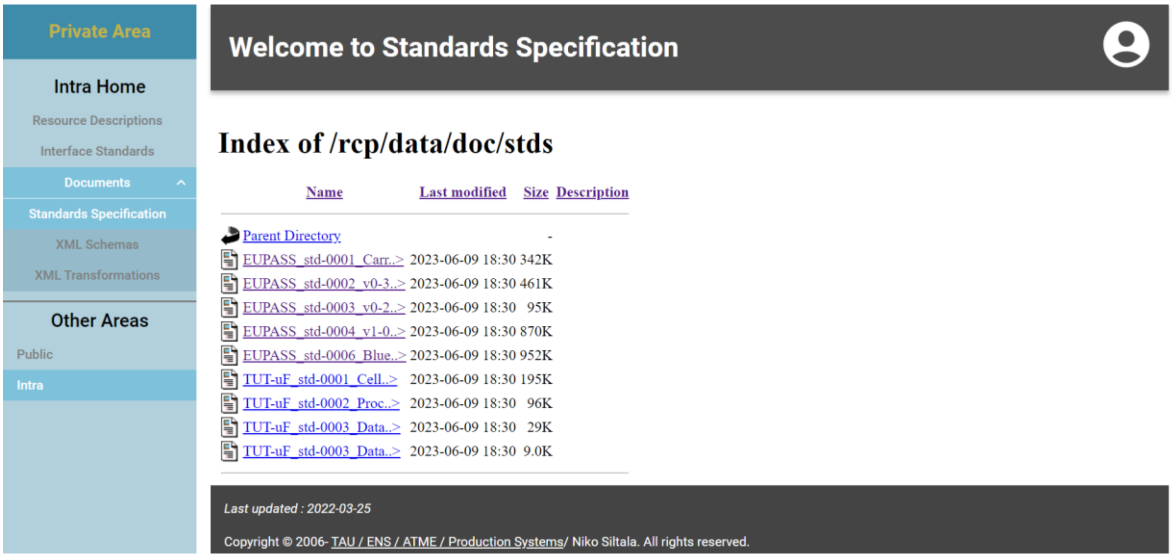

Figure 12: User Interface of Standards Specifications

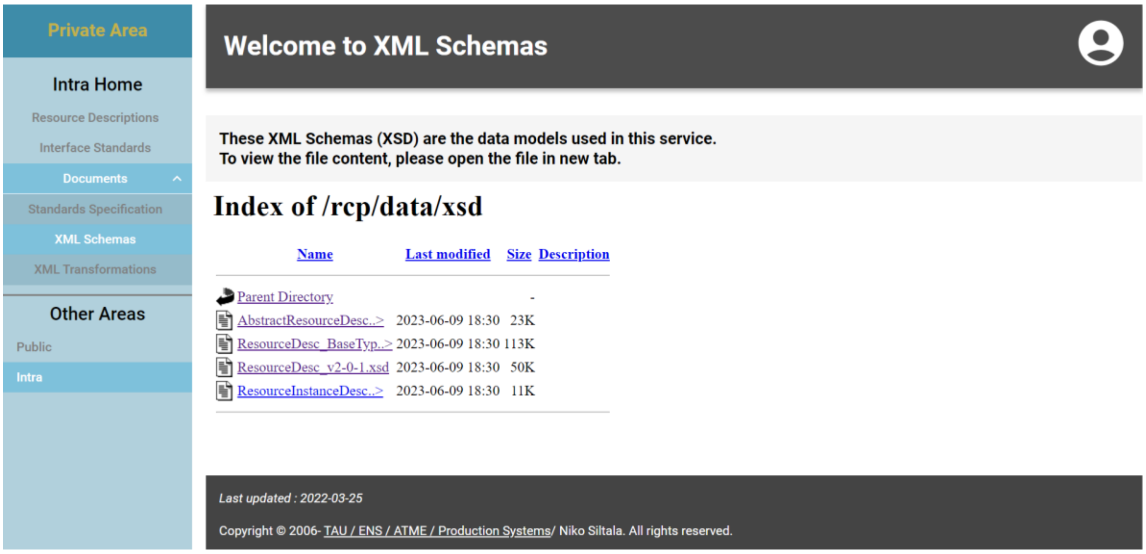

Figure 13: User Interface of XML Schemas

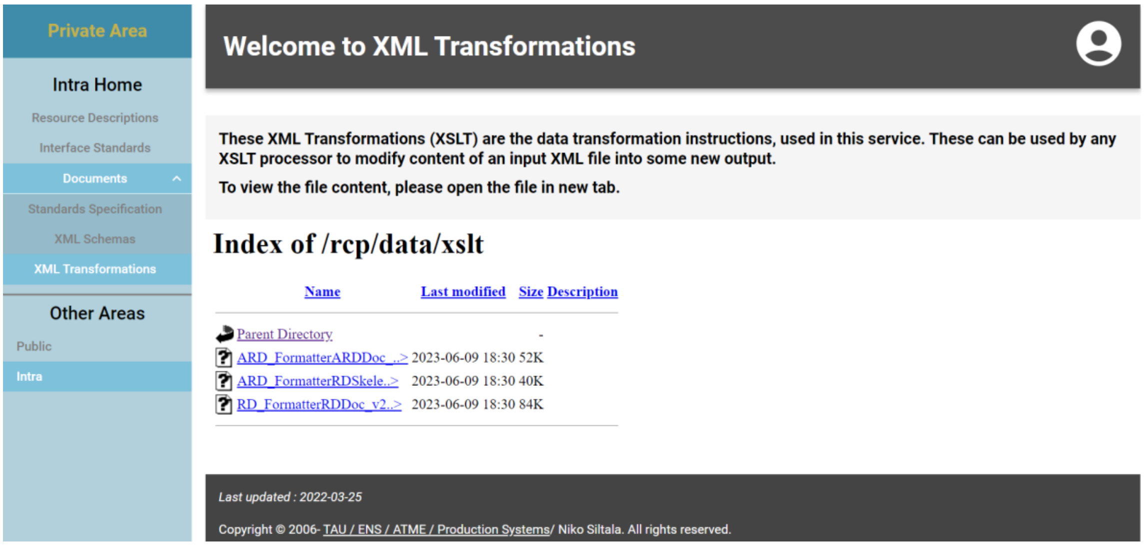

Figure 14: User Interface XML Transformations

4.4.3.2. Resource Provider Role

As mentioned earlier, it's crucial to note that users with the role of "*Resource Provider*" also have access to the interface available to regular *Users*. However, specific features unique to users in the "Resource Provider" role are detailed below, along with accompanying screenshots of their respective user interfaces.

4.4.3.2.1. Upload New Resource Descriptions

Resource Providers can seamlessly upload new Resource Description files in XML format by clicking the "Choose file" button within the given UI, as illustrated in Figure 15. Moreover, they have the capability to view and download the Resource Description, facilitated by clearly visible icons, as highlighted in Figure 15. The My Resources view shows only the RDs, which are uploaded i.e. owned by the logged in user.

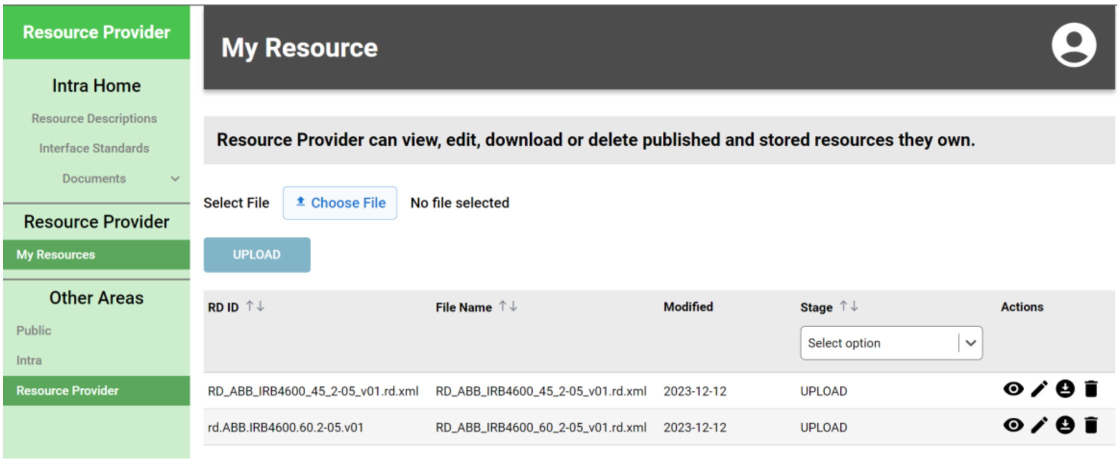

Figure 15: User Interface of Resource Provider's Resources

4.4.3.2.2. Filter the Resource Description

Important information for the Resource Provider is the stage of each individual RD. This defines the current stage for the description from upload to published. Resource Providers can efficiently filter the list of RDs they have uploaded based on various 'Stage' filter options. The available filtering options for RDs according to their Stage are UPLOAD, TO_BE_REVIEWED, IN_AUTOMATIC_VALIDATION, REVIEW_RETURNED, IN_REVIEW, REVIEW_APPROVED, PUBLISHED, and WITHDRAW, as illustrated in Figure 16. Each of these Stage options holds a specific meaning to the RD publishing process.

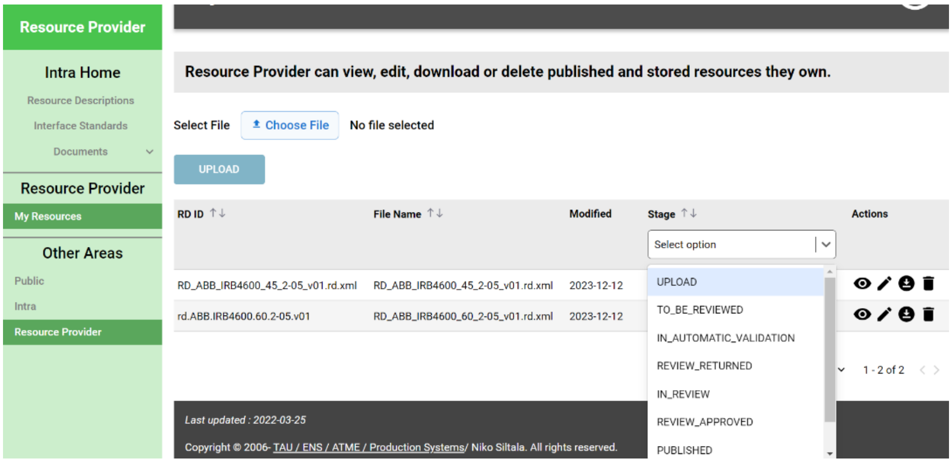

Figure 16: User Interface of filtering Resource Descriptions based on Stage Options

4.4.3.2.3. Edit and Delete the Uploaded Resource Description

Resource Providers possess the capability to edit or delete Resource Descriptions based on their needs, but this authority extends exclusively to the RDs they have uploaded. The editing process goes beyond mere modifications to the RD file; it also includes the attachment of connected data, which may consist of related additional documents or image files. The Edit interface is easily accessible by clicking the "Pen" (Edit) icon, as depicted in Figure 15. Once within the edit interface, Resource Providers can seamlessly attach related files using the "Append Files" section, accessible by clicking it (Figure 17).

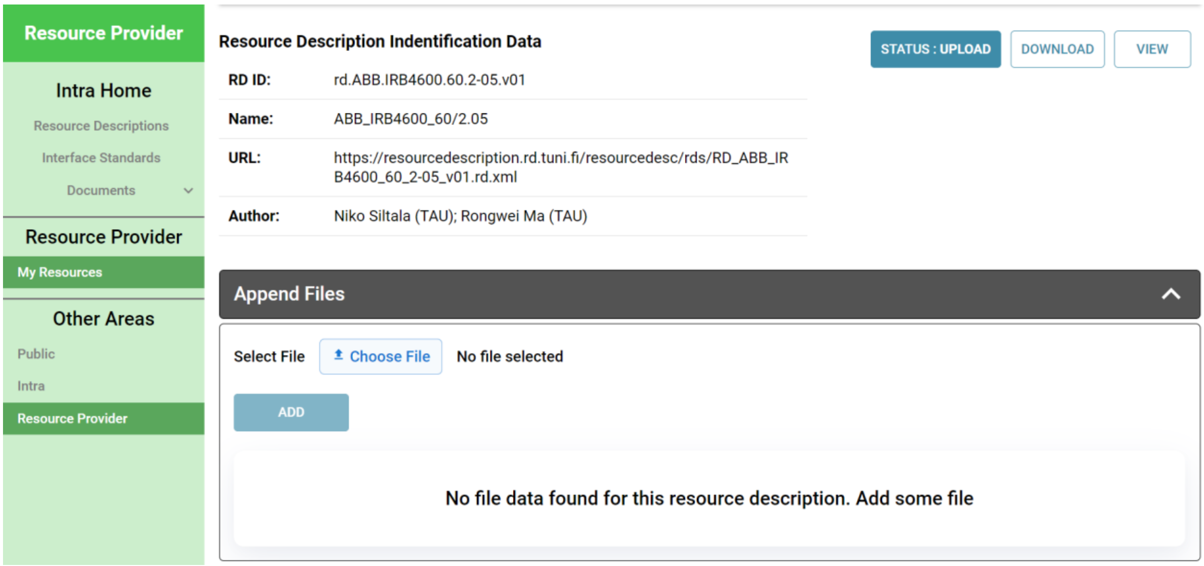

Figure 17: User Interface of Attaching RD's connected data

If deleting the Resource Description, the interface issues a warning message to the user, emphasizing the irreversible nature of the deletion as shown in Figure 18. It's crucial to note that the Resource Provider cannot delete an RD without first deleting its inner attachment(s) as in Figure 17. If deletion is done while having inner connected data, then the interface will show an error message (Figure 19).

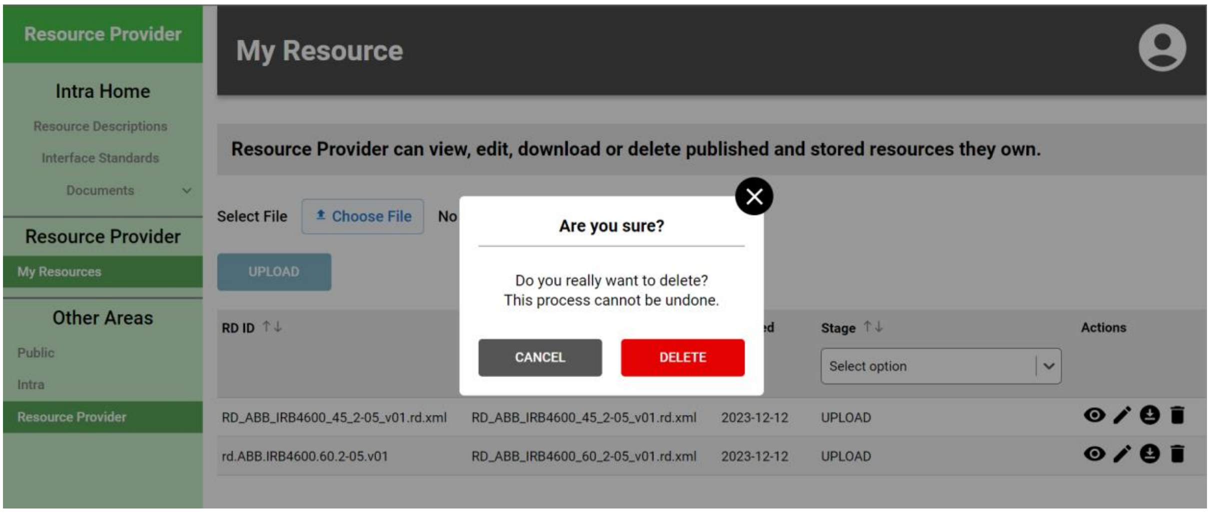

Figure 18: User Interface of warning message when deleting RD

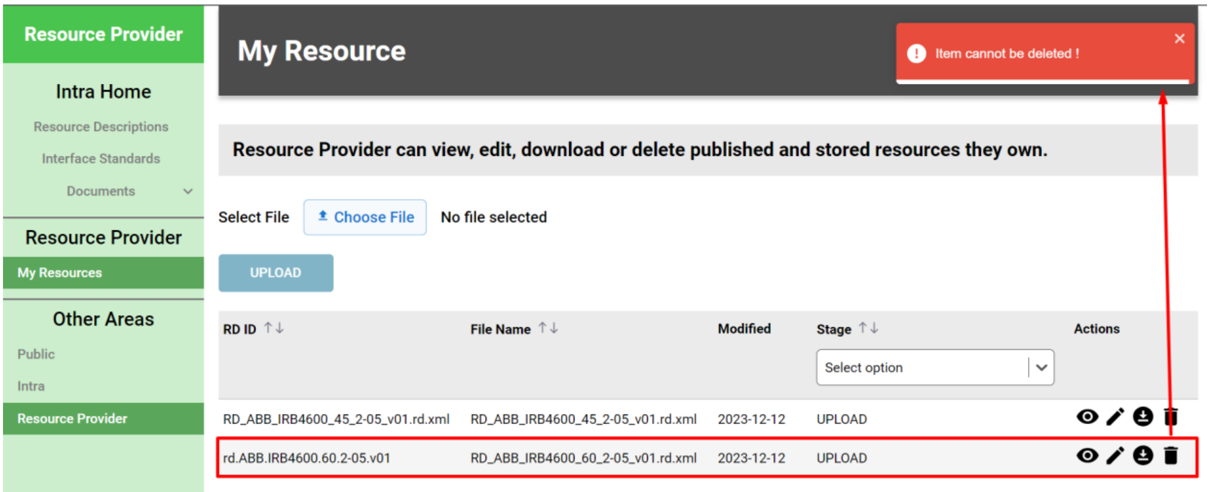

Figure 19: **Error message notification when deleting a RD with Inner Attachments**

4.4.3.2.4. Update, Validate, Review Resource Descriptions

After adding a Resource Description, Resource Provider can upload and update it by accessing the edit interface and selecting 'Update and Validate Resource Description,' as demonstrated in Figure 20. Updates can take place e.g. when RD has some updated data, usually links. Need for link update can take place once appended files are uploaded to RCP service, and URL reference for those files becomes available.

Additionally, the UI has the option to validate the Resource Description (postponed for future implementation) and view the validation report (postponed for future implementation) by clicking the respective buttons. Validation would check the submitted RD against the data model of RD (XSD), and result of this process will be shown as validation result report. This will capture issues such as missing or wrong data structure(s); missing of mandatory elements; data type mismatch. In the absence of a validation report, the interface will display a notification message about its unavailability.

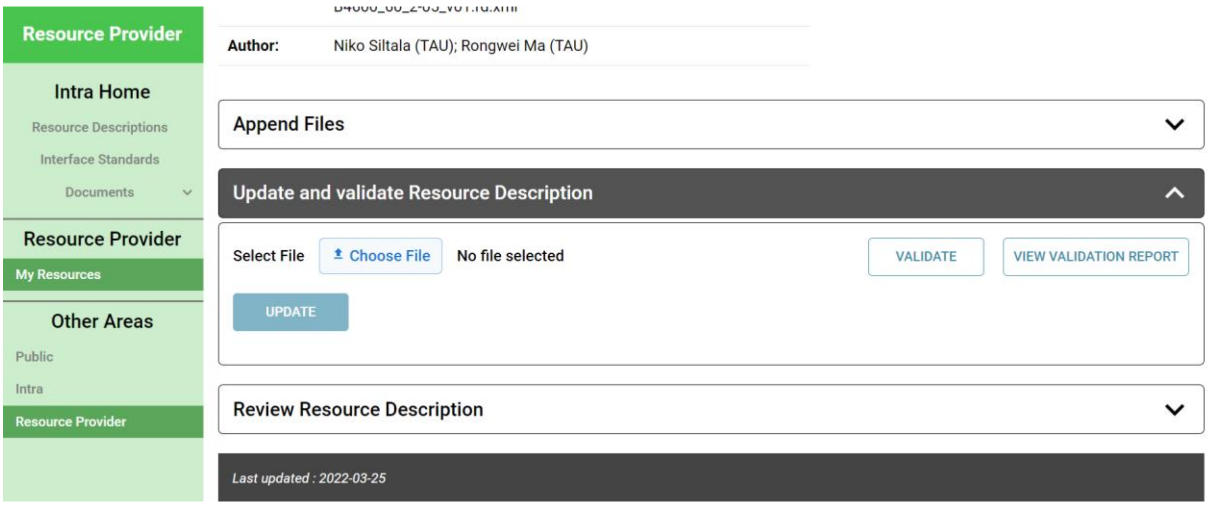

Figure 20: **User Interface of updating and validating Resource Description**

Similarly, after uploading and finalizing a Resource Description, Resource Providers have the option to submit it for review by the Resource Managers. This can be done by entering the edit RD UI, navigating to the 'Review Resource Description' section, and clicking the 'Submit to Review' button (postponed for future implementation) as shown in Figure 21. Additionally, the UI has the option to view the review report, if available, by clicking the 'View Review Report' button (postponed for future implementation). The review report provides feedback for Resource Provider from Resource Managers about the RD.

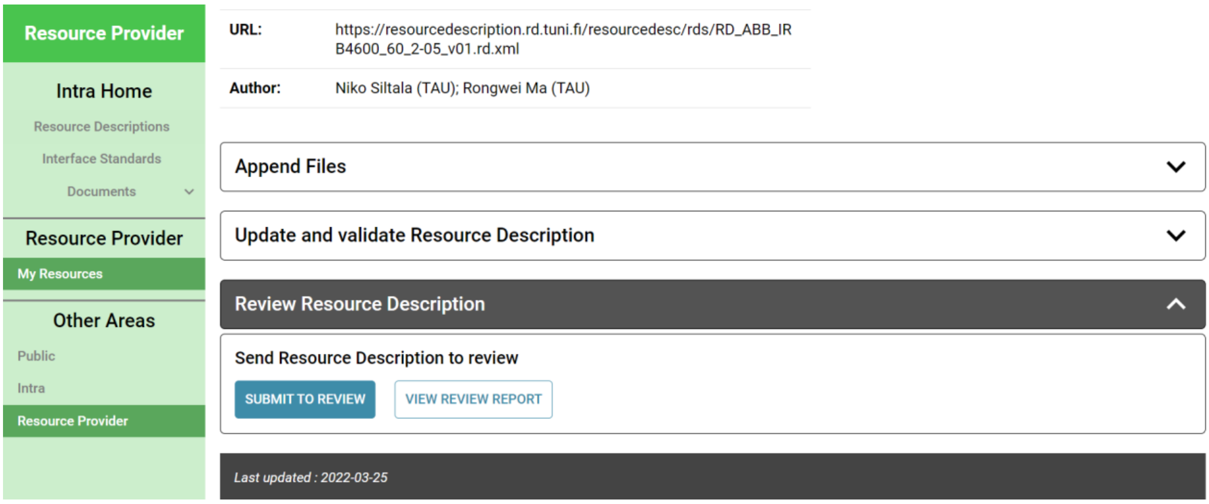

Figure 21: Submitting Resource Description for Review and Accessing Review Report

4.4.3.3. Resource Manager

As mentioned earlier, it's crucial to note that users with the role of "*Resource Manager*" also have access to the interface available to regular *users*. However, specific features unique to users in the "Resource Manager" role are detailed below, along with accompanying screenshots of their respective user interfaces.

4.4.3.3.1. View and Filter the Submitted Resource Descriptions

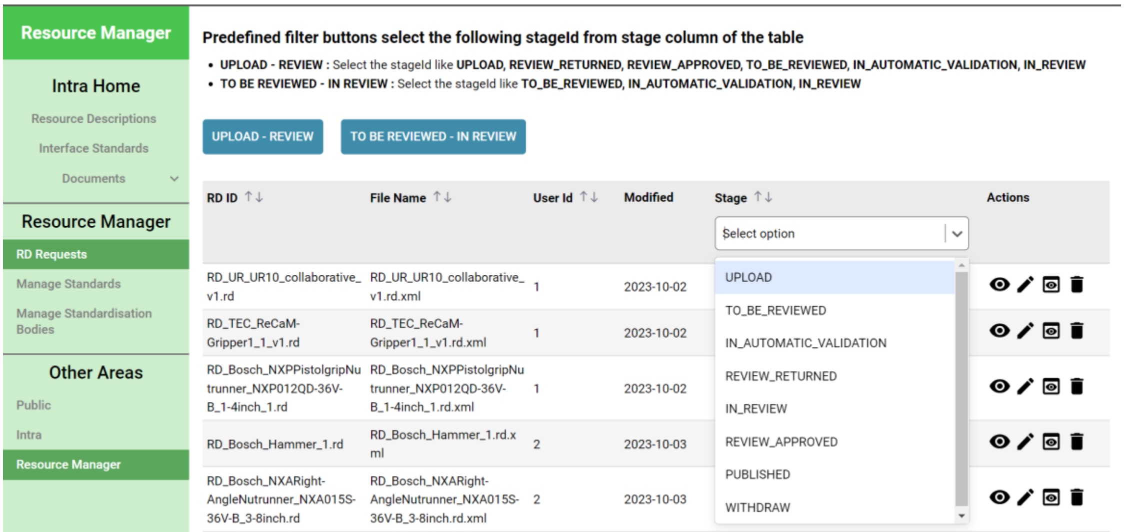

Figure 22: User Interface of Filtering RD by Resource Manager

The Resource Manager dashboard comprises a comprehensive list of all uploaded Resource Descriptions by various Resource Providers. Resource Managers have the flexibility to manually filter RDs based on their stage by choosing from the option list, as demonstrated in Figure 22. Multiple stages can be selected simultaneously. Alternatively, they can use the convenient global buttons, 'UPLOAD - REVIEW' or 'TO BE REVIEWED – IN REVIEW,' accessible in Figure 22. These buttons directly initiate filtering and provide details about the specific stage selected from the option list upon activation.

4.4.3.3.2. Edit, and Delete the Uploaded Resource Descriptions

Resource Managers have the capability to edit and delete Resource Descriptions, following procedures outlined in Figure 17 and Figure 18. While they cannot upload new resource descriptions (not in this view and role), they possess the authority to update existing ones uploaded by other Resource Providers.

This update capability allows them to rectify omissions, add missing information, or enhance connected and harmonized data associated with a particular RD. Resource Managers can seamlessly perform all functionalities available in the edit interface, like the interface used by Resource Providers.

4.4.3.3.3. Review, Approve, Reject Resource Description

The reviewing process for a resource description commences when a Resource Provider submits the RD for review (refer to Figure 21), and the Stage of that RD changes to 'TO_BE_REVIEWED.' Accessing the review UI (Figure 23) can be done by clicking the review icon (middle icon), as depicted in Figure 22. Upon initiation of the review, the Resource Manager can change the Stage status by clicking the 'Start Review' button (postponed for future implementation). Within the review interface, Resource Managers provide feedback on the specific RD file, with a feature to save their feedback. After writing feedback, they can choose to either approve the Resource Description for publication or reject it as shown in Figure 24 (postponed for future implementation). The review report detailing the approval or rejection is then made available to the respective owner of that resource description.

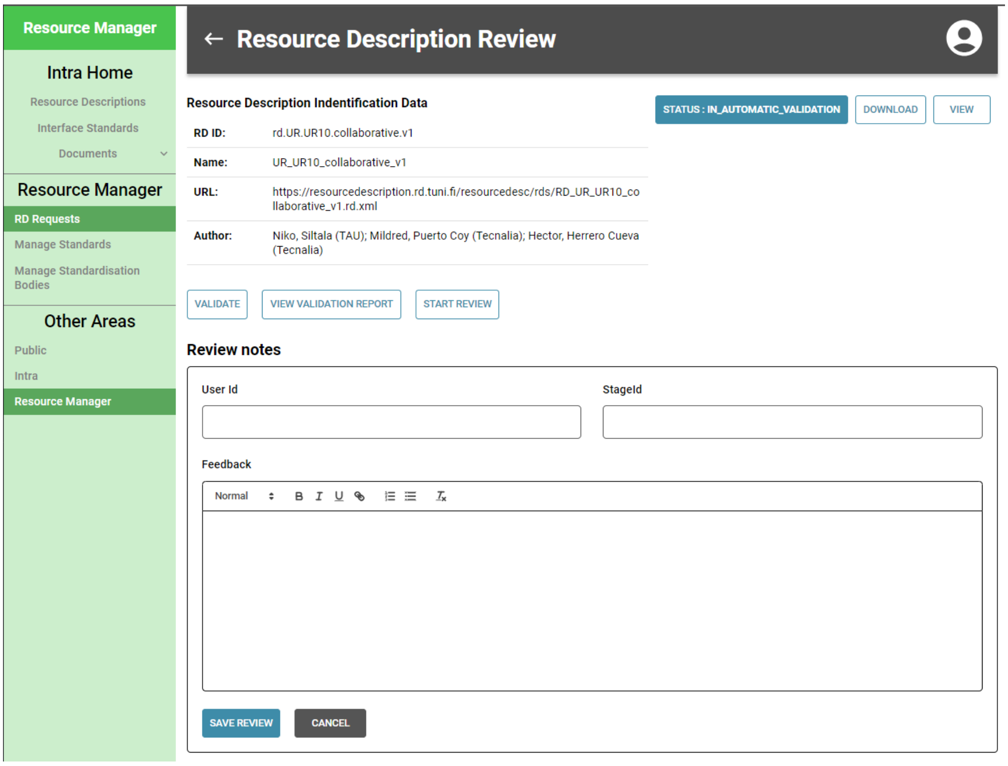

Figure 23: User Interface of Resource Description review by Resource Manager

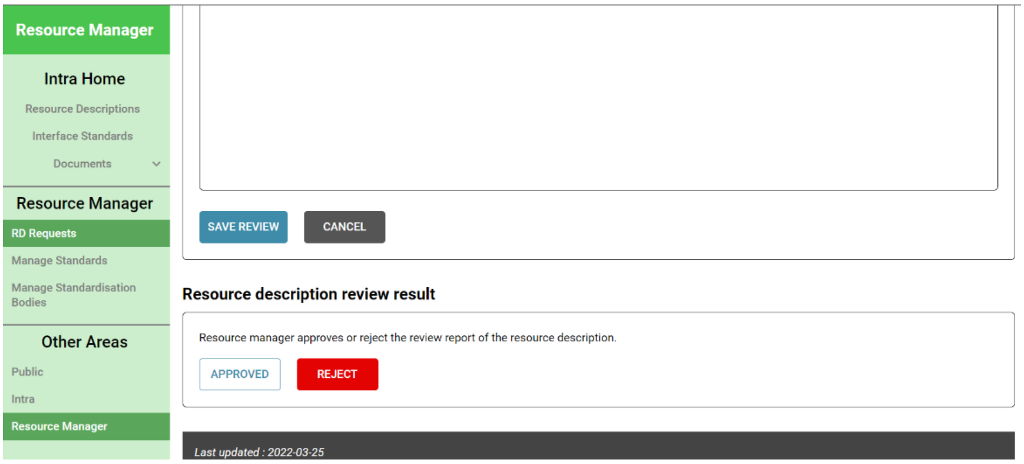

Figure 24: User Interface of approving or rejecting Resource Description

4.4.3.3.4. Add Interface Standards

The Resource Manager interface includes a comprehensive list of Standards, as illustrated in Figure 25. In addition to managing existing Interface Standards, Resource Managers hold the authority to add new Standards and modify the existing.

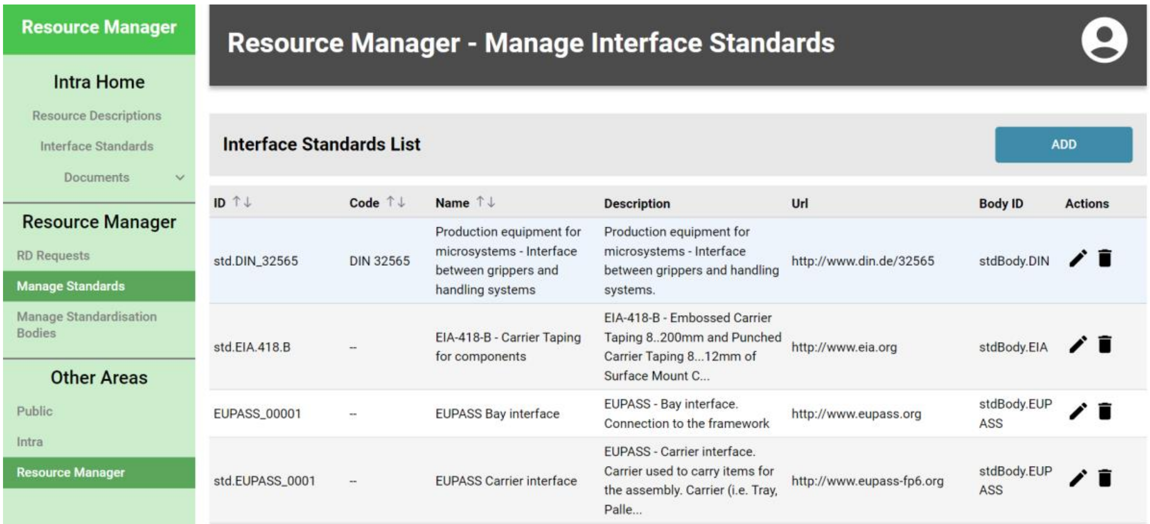

Figure 25: List of Interface Standards in the Resource Manager Standards UI

To add a new Interface Standard, Resource Managers can simply click the 'ADD' button located in the top right corner of the UI. This action opens a popup section, as illustrated in Figure 26, where they can conveniently fill out the necessary information and add it by clicking the "ADD" button on the form.

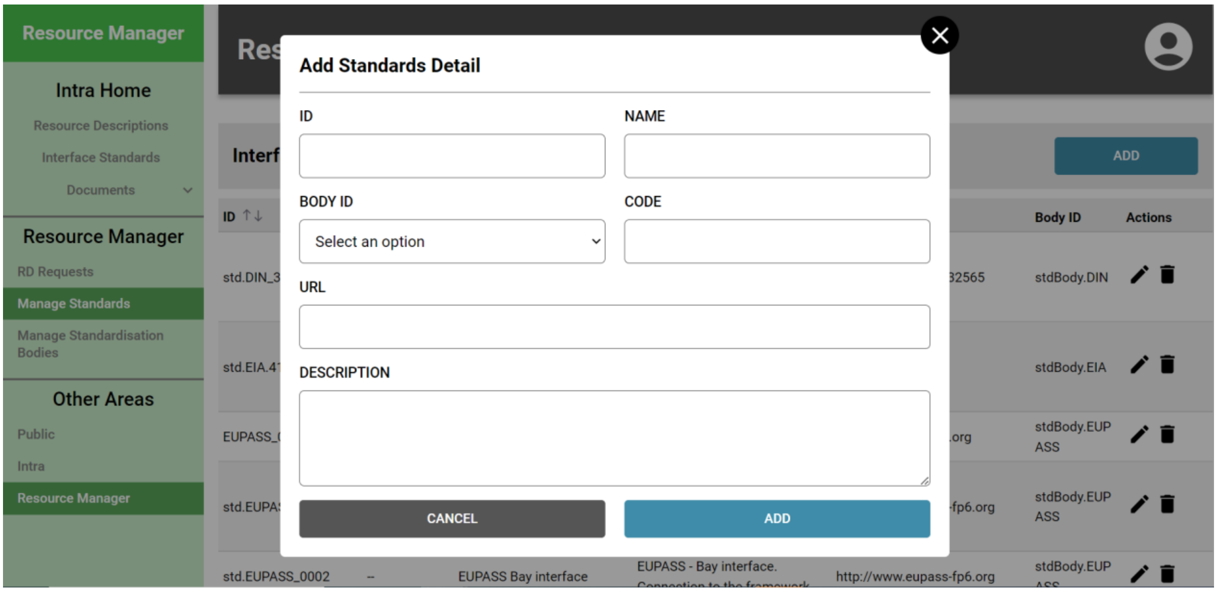

Figure 26: User interface of adding new Interface Standard

Upon successfully adding a new standard, a notification message will appear confirming the successful addition (postponed for future implementation). Conversely, if a Resource Manager attempts to upload without filling the required information fields, the interface will display an error message pertaining to the incomplete field, as depicted in Figure 27.

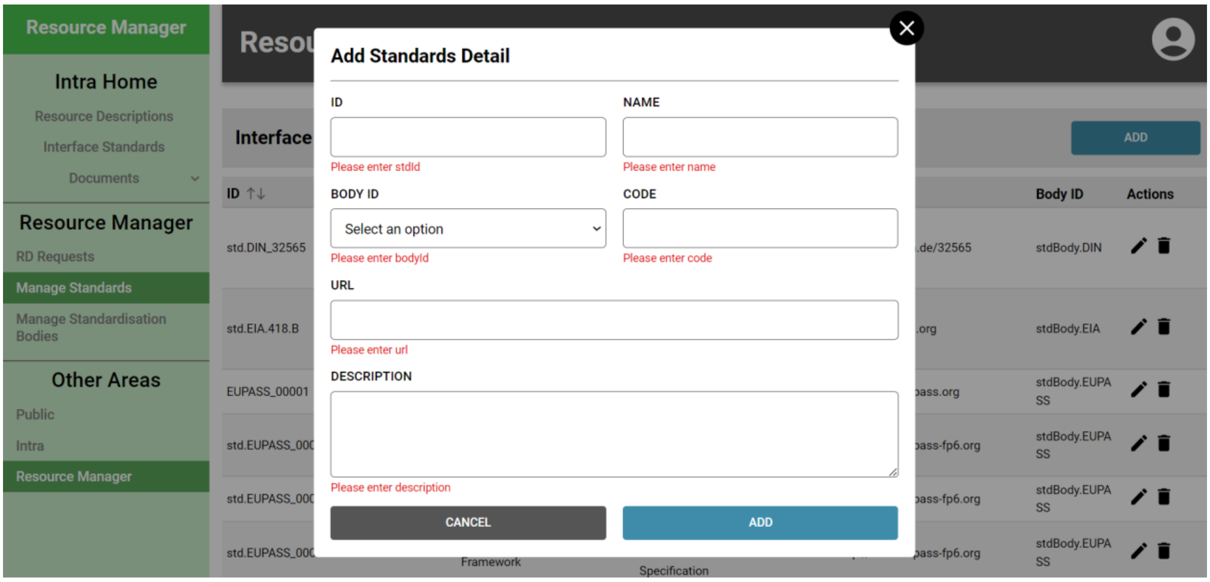

Figure 27: Notification of Error Message for Incomplete Information

4.4.3.3.5. Edit, and Delete Interface Standards

Resource Managers can effortlessly edit or delete existing Standards within the interface using the corresponding icons. The deletion process triggers the same warning message depicted in Figure 18. To edit a Standard, Resource Managers can simply click the edit icon (pen) as displayed in Figure 25. Upon clicking the edit icon, a popup window appears (Figure 28), presenting the existing information for each field. This allows Resource Managers to easily edit the required information and update it to make the changes visible (postponed for future implementation).

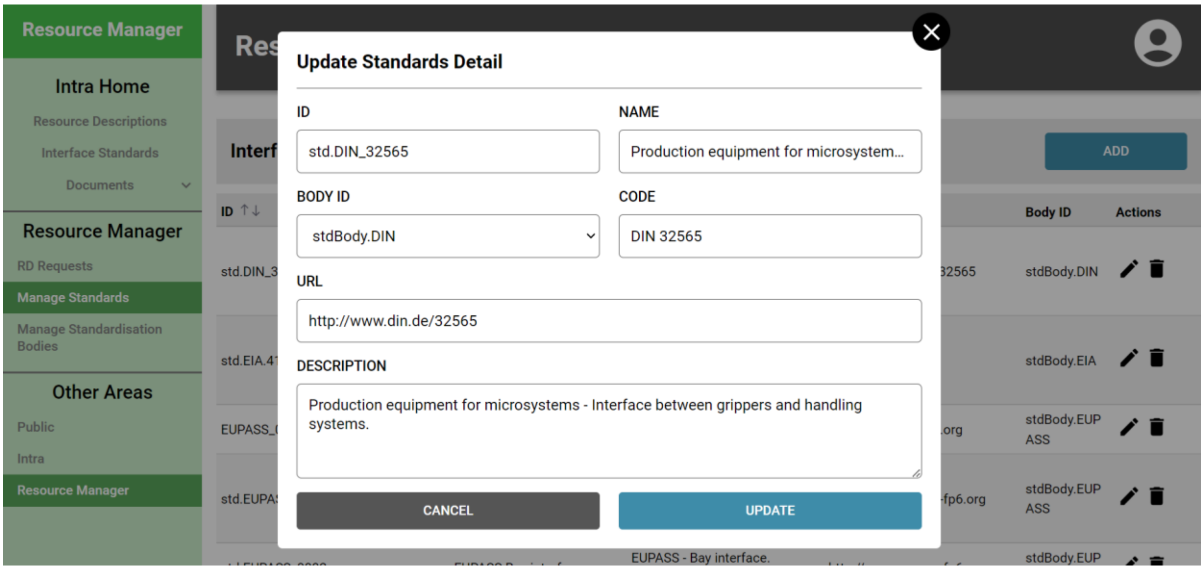

Figure 28: User Interface of editing Standard's Information

4.4.3.3.6. Add, Edit and Delete Standard Bodies

Similar to Interface Standards, Resource Managers also manage a list of Standard Bodies, depicted in Figure 29. Resource Managers can seamlessly perform similar functionalities with Standard Bodies—such as adding new bodies (Figure 30), editing existing entries (Figure 31), and deleting entries as needed.

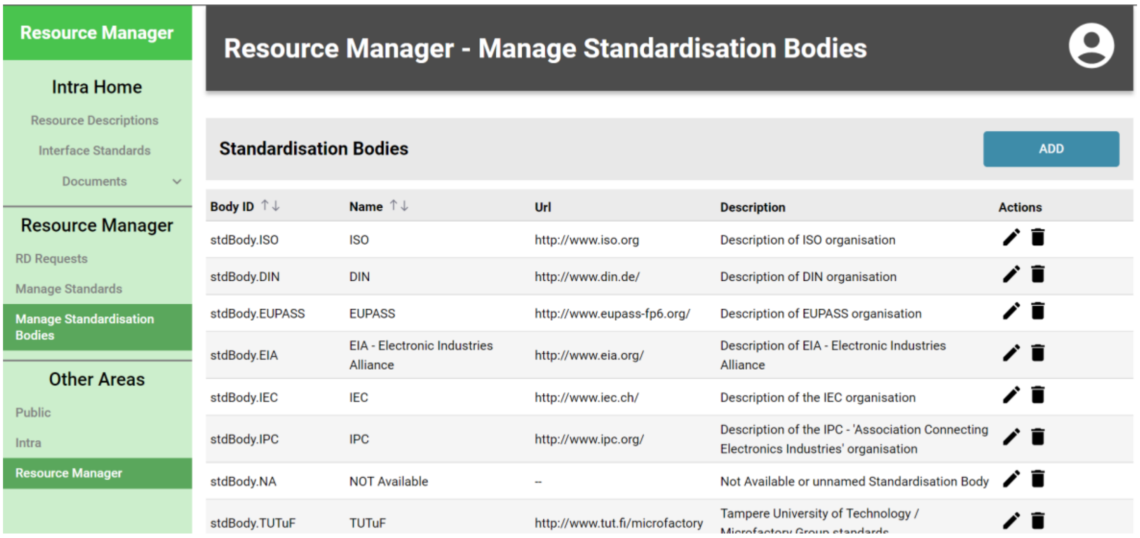

Figure 29: List of Standard Bodies in the Resource Manger Standard Bodies UI

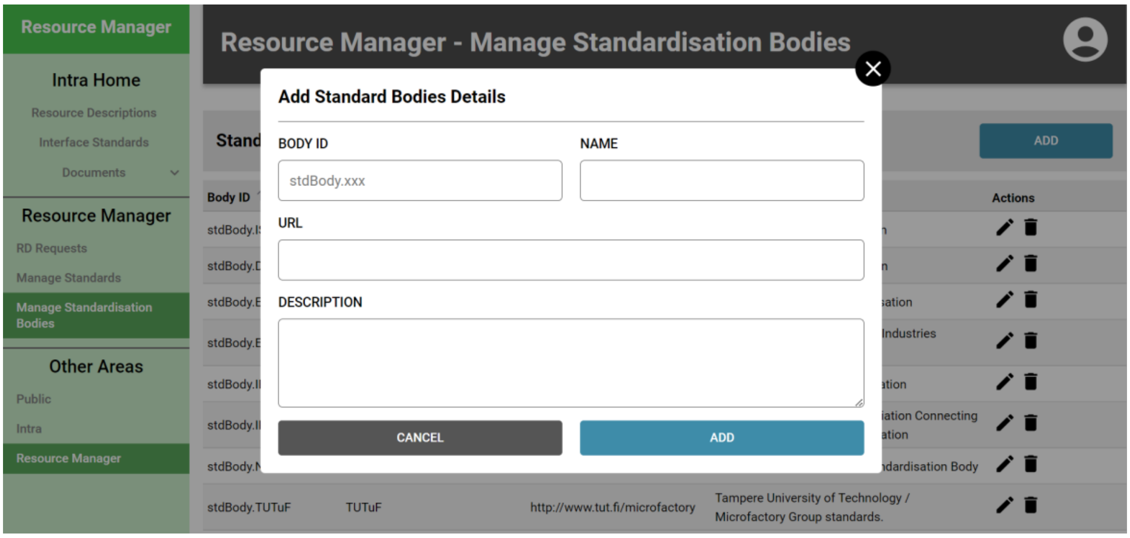

Figure 30: User interface of adding new Standard Bodies

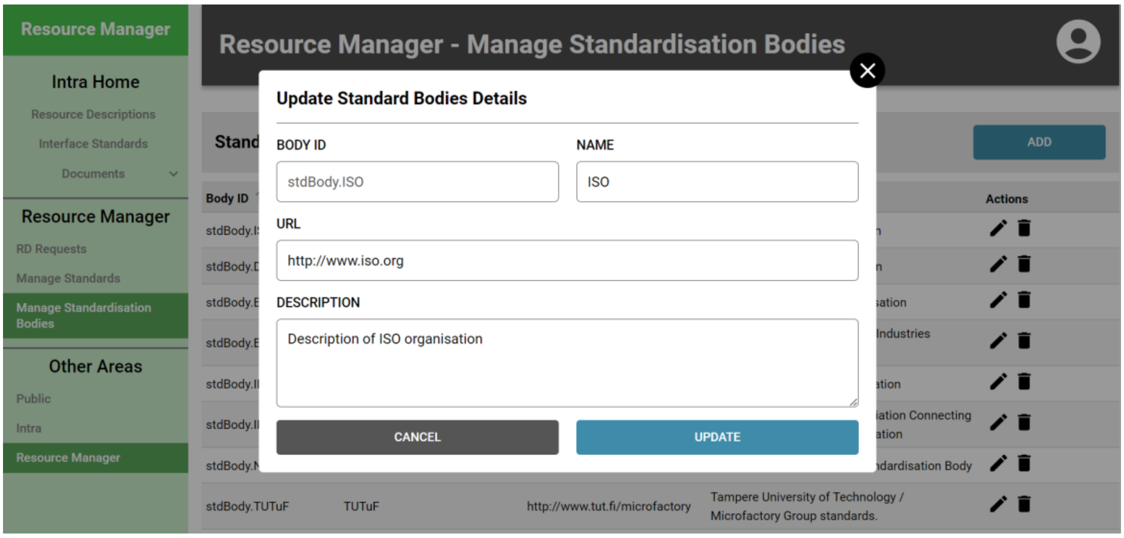

Figure 31. User Interface of editing Standard Bodies' Information

5. Digital Twin

5.1. Overview

KTH has proposed an operating system of the digital twin to connect virtual and real HRC cells. We built an HRC system implementing digital twin functionality which includes several components for Java 3D models of human-robot work cells, sensor data processing and fusion, and communication between the physical and digital twins. For network transport, the Digital Twin system uses a Transmission Control Protocol for the communication between the robot and the server, while HTTP streams are used for data sharing from the server to remote users. Sensor data from the robot are passed to the server in packets when a human operator opens a subscription to robot monitoring for a specific work area. The visualization includes not only the pose information and operation patterns of the monitored robot but also the movements of the human operator after being calibrated, ultimately enabling communication and connectivity for HRC in both virtual and physical spaces (Figure 32).

On the other hand, a digital twin system based on the software of Visual Components is designed to reflect the real-time motions of human operators and robots. For the human operators, the digital human model with predefined hierarchy is set up in the software. The data collection hardware communicates with the software in real time via socket to synchronize the movements of physical operators with the digital operators. For the robots, the ABB IRB 120 and ABB IRB 1600 robot models are configured with associated communications between these models and controllers of physical robots. Figure 33 below shows the overall scene of digital twin.

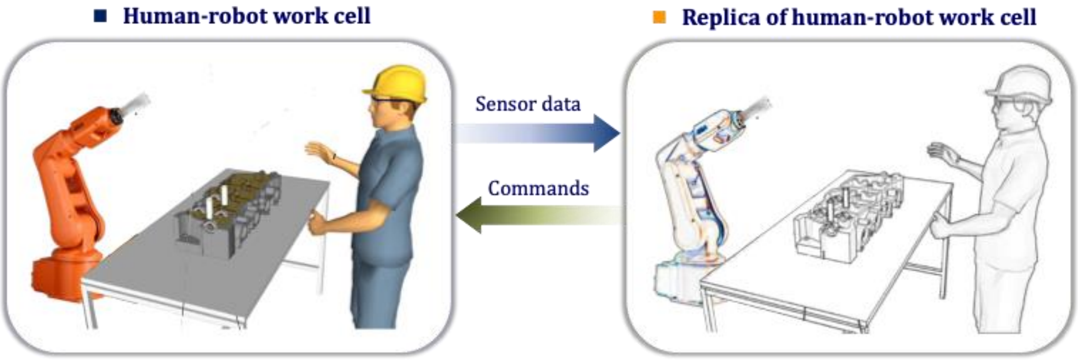

Figure 32: The general representation of the digital twin for HRC cells

Figure 33: The digital twin of human operator and robot in Visual Components 4.0

5.2. Overall methodology

The Kinect V2 sensor is used to detect the skeleton of human body with the structure shown in Figure 34 below. The skeleton consists of 25 joints for each individual human body. The 3D position, 3D orientation, tracking state and restriction are recorded for each body joint. While it is sometimes difficult to track the occluded joints or part of the body skeleton, Unscented Kalman Filter (UKF) is specially designed to improve the consistency of joint coordinates in the continuous movements of human body (as shown in Figure 34). After filtering with UKF, the noises underlying the measurements of joint coordinates can be significantly reduced. Moreover, deep learning-based methods have been investigated to estimate the 3D coordinates of 25 body joints. Figure 36 and Figure 37 show some of the results, where the occlusion issues are partially addressed.

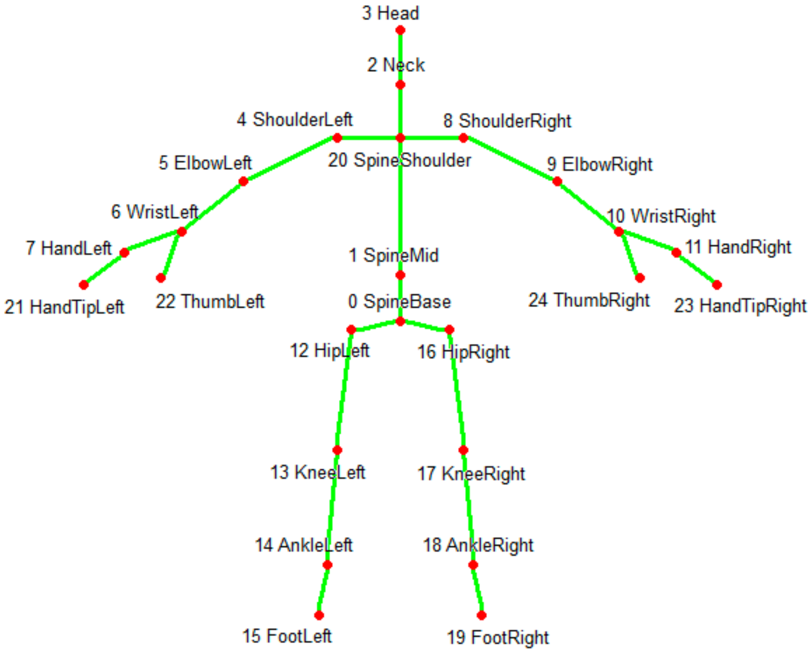

Figure 34: The structure of human body skeleton detected by Kinect V2

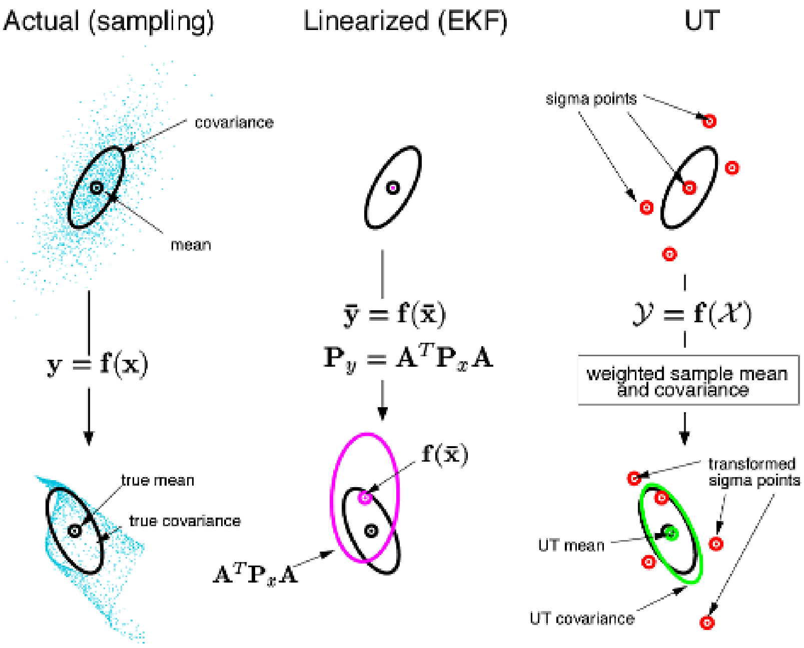

Figure 35: An overview of Unscented Kalman Filter (UKF)

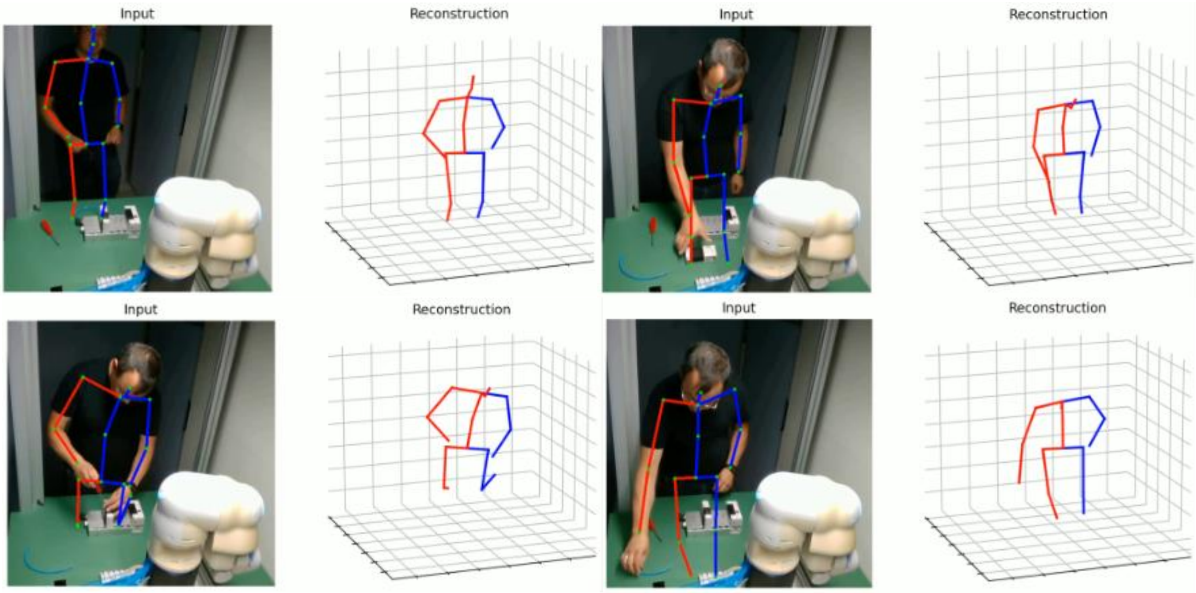

Figure 36: Some detection results by a Transformer-based model

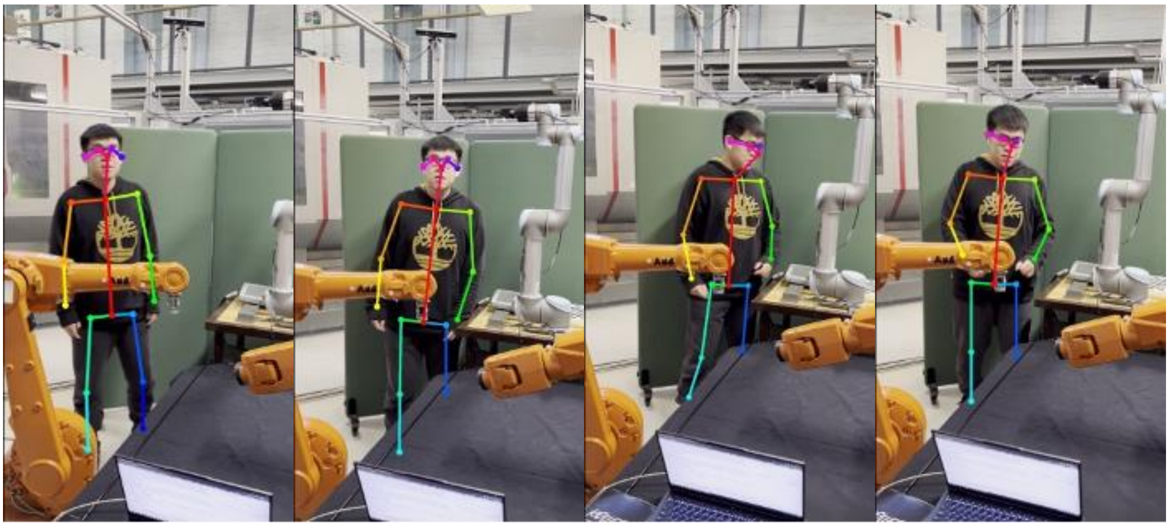

Figure 37: Some detection results by the OpenPose estimator

In terms of the mapping from the detected human body skeleton to the ongoing motions of digital operator in the Visual Components software, a combination of forward joint angle calculation and inverse kinematics is designed to improve the robustness of task-oriented entire movements. In terms of the forward joint angle calculation, the bones are first represented as spatial vectors based on the coordinates of detected 25 key points (joints). Then the angle of two vectors along a specific axis is calculated by projecting the vectors into the plane normal to that axis and using the arccos function to derive the value. Figure 38 below shows a brief mechanism of such calculation. Based on these calculations, some of the necessary angles are transferred to the digital operator in the Visual Components. A threshold of rotation range is set for each joint individually, so that the movements of digital operator adhere to the physical constraints of human body. In terms of the inverse kinematics, it is implemented for the upper part of human body. Backward calculations are performed from the spine to the joints on the left and right arms. The mechanism of calculation is similar to the inverse kinematics for the robot arms. Moreover, during the normal movements only forward joint angle calculation is enabled to drive the digital operator. When the human approaches certain objects and shows the intention to interact with them, the inverse kinematics will be triggered.

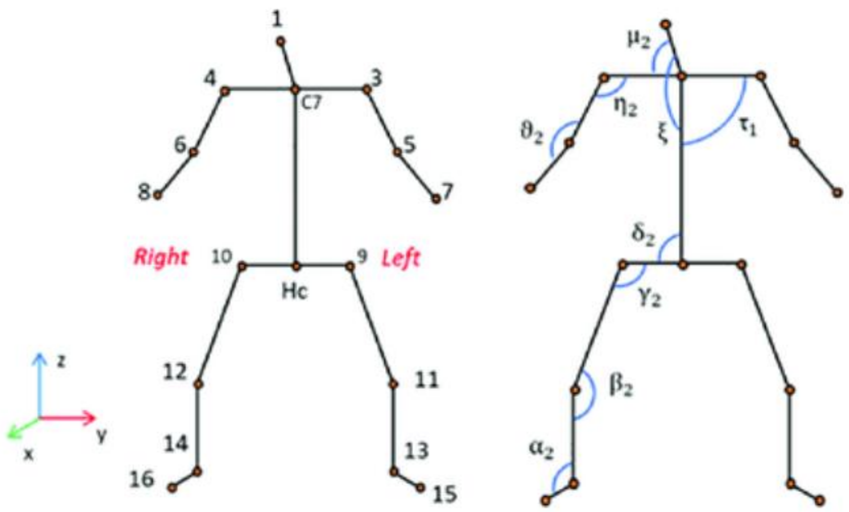

Figure 38: A brief illustration of the joint angle calculation based on detected human body skeleton

5.3. Integration of digital twin with other ODIN tools

The connection of Digital Twin with other modules of the Digital and Networked components was required for its realization. In its final version, the Digital Twin module is directly connected with the OpenFlow for the required data exchange with the rest of the modules. In more details, the Digital Twin of ODIN is integrated with the following modules:

- OpenFlow

As already mentioned, the Digital Twin module is directly connected with the OpenFlow. The OpenFlow acts as the main communication channel between the Digital Twin and the other modules of ODIN. Thanks to OpenFlow, the Digital Twin is always up-to-date regarding the latest pose of each robotic resources as well as the result of object detection module's. Additionally, thanks to PILZ laser scanners' integration with ROS, the Digital Twin is updated on-the-fly about safety areas shape.

- Digital Simulation

The Digital Simulation module is of high importance for the Digital Twin module as it is used to visualize the real-time data of the DT. The robot and sensor models included in the library of the Visual Component software enabled the easy visualization of the investigated pilots DT. The integration of the DT module with the Digital Simulation has been achieved through the implemented ROS framework.

- AI Task Planner

As presented in deliverable D3.3, the Digital Twin of ODIN is integrated with the AI Task Planner module for the initialization of the Digital Simulation layout during the digital validation of the generated task plans. The Digital Twin provides the required data for the resources initialization in the Digital Simulation and it is of high importance in case that the task planner is triggered in its online mode and the exact position of the resources in the digital simulation may effect the evaluation data.

5.4. Final Implementation

The digital twin system is implemented first in the KTH pilot case with sufficient testing, and then deployed in the other pilot cases successively (starting from the White Goods pilot case). The testing and fine tuning of digital operator in the digital twin system is of paramount importance, as the physical human operator usually shows complicated movements when performing a certain task. The digital operator should be able to represent these movements by supporting higher degrees of freedom for the joint rotations. Besides, the external factors such as occlusions should be considered. Figure 39 below shows some of the testing results in the KTH pilot.

Figure 39: Some results of motion synchronization between physical operator and digital operator

The collaboration between human operator and ABB IRB 120/ABB IRB 1600 is demonstrated & tested (Figure 40). The HRC system status is recognized in real time to facilitate the decision-making.

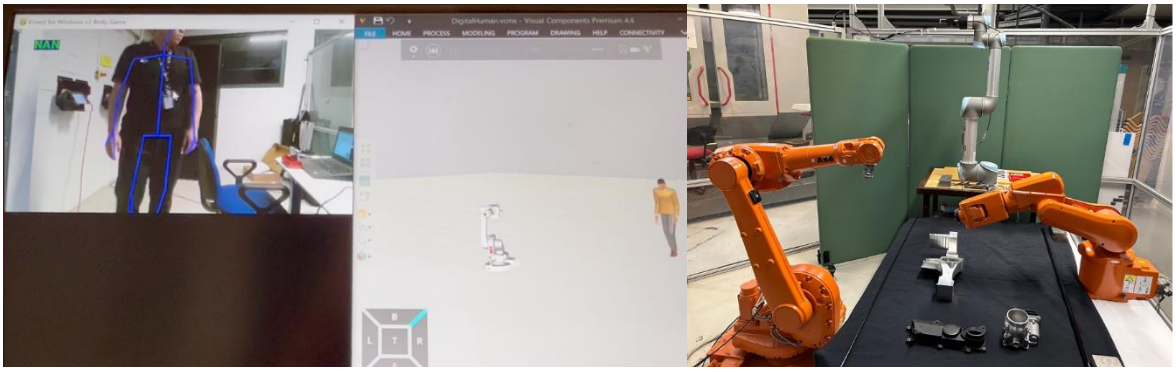

Figure 40: An overview of the physical HRC system and its digital twin system in KTH pilot case

6. DIGITAL SIMULATION

6.1. Overview

Digital simulation in manufacturing is a technique that uses computers to create and analyse virtual models of real-world manufacturing plans, cells, systems, devices, processes, or tasks. Digital simulation can be used for various purposes, along the entire manufacturing system life cycle phases (Figure 41) such as: conceptualize, design, deploy and commissioning, operate and optimize using the digital twin, retrofit for new production requirements and reuse in new production lines during the decommissioning phase.

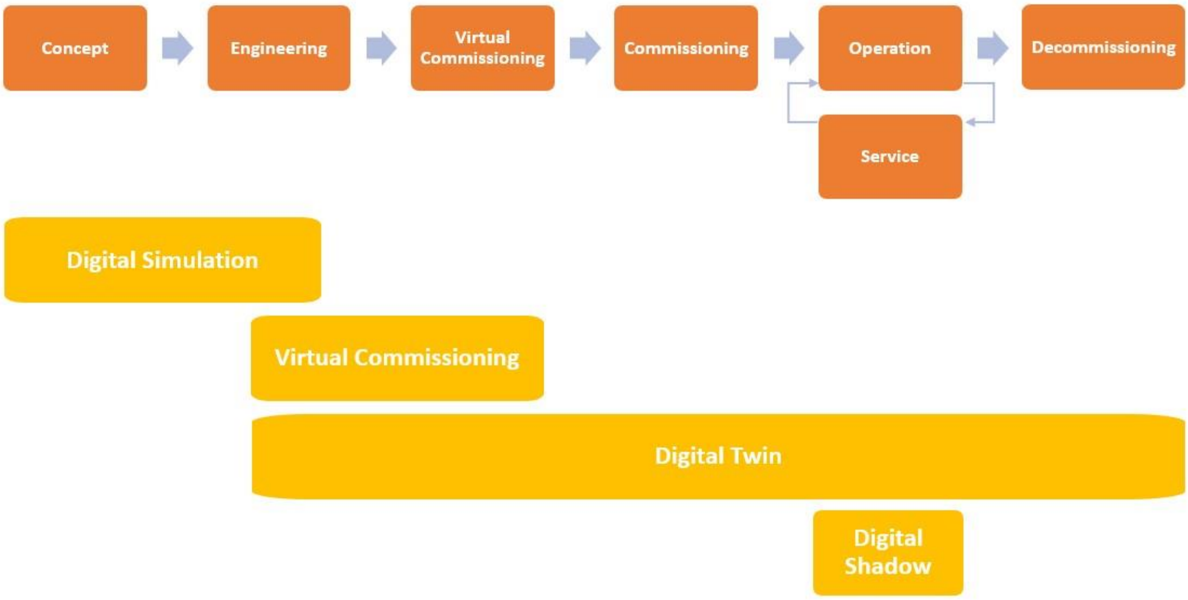

Figure 41: Phases in the manufacturing system(s) life cycle

Despite the work developed in ODIN is focused on the process verification, the decommissioning phase is not included in the research, nevertheless it is relevant to mention as from the development side is important the availability of digital models. The acquisition of operational data during the production phase and the maintenance of this data will facilitate the future reuse of the digital information in case of reconfiguration and reusability of the manufacturing systems available in the production line.

The digital simulation within ODIN pilots has been developed using Visual Components 4.0 (VC 4.0) and particularly the Process Modelling (PM) feature. Process Modelling is a feature included in Visual Components 4.0 which has been further developed within ODIN and simulation components has been created to reach the ODIN simulation requirements developing the ad-hoc simulation library for ODIN. During the project several releases of VC 4.0 has been used incorporating the advances in process modelling and interfaces developed within the project to be used in the project pilots. At the moment of writing this deliverable the release 4.8 (VC 4.0 (r.4.8)) is being used.

The three pilots developed within ODIN (Aeronautics, Automotive and White Goods) are using VC 4.0 and the simulation components as mentioned previously have been created using mainly process modelling from the initial phases, as starting point for creating the simulation, building the respective simulation models. During the design phase that includes the engineering phase these components have been extended adding additional behaviours to reach the expected pilot requirements.

6.2. Overall methodology

6.2.1. Component development

The base of the digital simulation is the virtual component (aka virtual model). The virtual component is the virtual representation of the real system in the virtual space provided by the digital simulation. The virtual component structure within VC 4.0 is depicted in Figure 42.

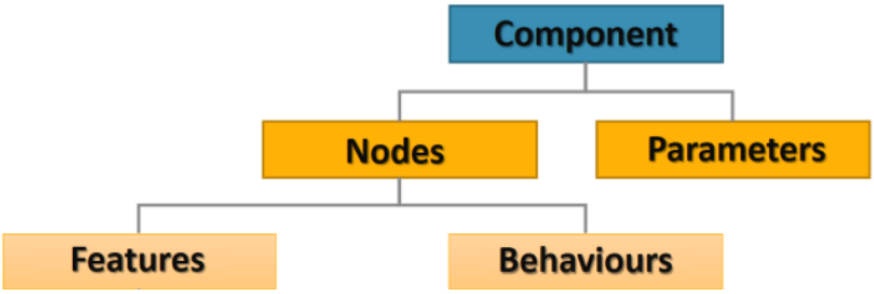

Figure 42: Virtual component structure in VC 4.0

Parameters define the properties that can be linked to features and behaviours. Properties are used in the Python API for creating the scrips that define the component simulation behaviours. There are several types of *properties*, such as real, integer, Boolean and distribution values, as well as string of characters. These types can be used when defining component dimensions, processing times, product filtering, failure times and probabilities, and other attributes of the component.

Features refer to frames, geometry, and other related features. Virtual components can include frames in their nodes, to define their location and orientation from a root node, as well as to coordinate material flow. Geometry defines the physical shape of the virtual component. Other related features include modification capabilities for transforming and switching.

Behaviors define the way the virtual component behaves for positioning (containers and paths), control (signals), sensors, interfaces between components, servos and actuators, kinematics and controllers, executors, physics, statistics, and user defined busing the python interface (scripting).

The complexity of the virtual component can be increased extending the granularity to mirror the real system and represent it in the virtual space integrating the required functionalities. Figure 43 details in a high level a component with different links. This allows to increase the level of detail to be simulated. To facilitate the development of the simulation components and the pilot's simulation the PDCA (Plan-Do-Check-Adjust) methodology has been followed.

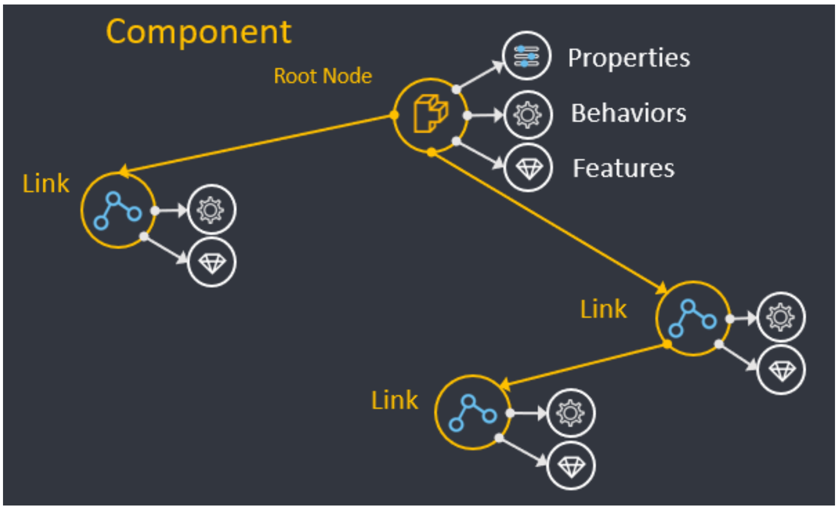

Figure 43: Detail of the building of a complex of the complex component

Introduced in D3.1 the PDCA approach is defined at [12] as an iterative design and management method used for the control and continual improvement of processes and products. The utilization of this approach while creating the pilots´ digital simulation models facilitates its implementation, as it provides an incremental way of work and can be used during the three phases considered within ODIN (Figure 44), of the system life cycle for the digital simulation, starting from the simplest component and extending it to the pilots layout.

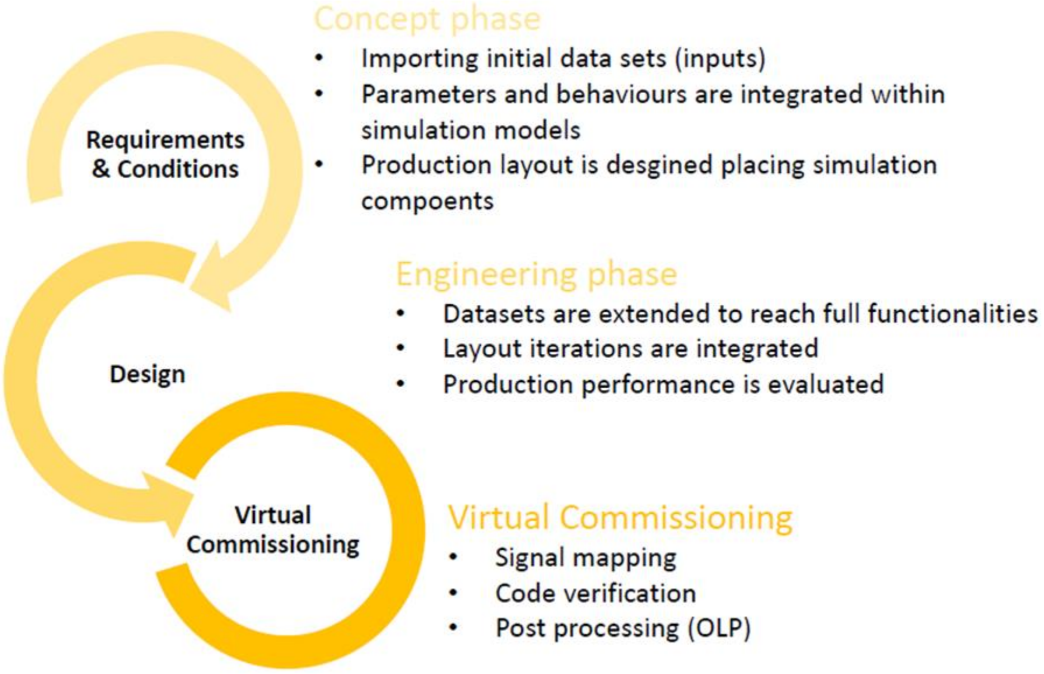

Figure 44: Detail of the phases concept, design, and virtual commissioning phases implemented in ODIN

This deliverable focus on the work completed in the concept phase and engineering phase, leaving the virtual commissioning to be covered in more detail in D3.4. Nevertheless, it is necessary to remark that virtual commissioning relies on the virtual component development for its implementation, as the behavioural signals, interfaces, actuators, and servos as well as the ad-hoc behaviours determine how the virtual commissioning phase is implemented.

The concept phase starts with the pilot requirements compilation, including functional and operational ones. Existing commercial equipment available within VC 4.0 electronic catalogue (eCat) can be used directly in the virtual environment. Non available equipment can be modelled starting from the parameters which define it. CAD data is imported to define features and add behaviours which conforms the virtual component to match functional and operation requirements. The manufacturing systems can have different levels of complexity (Figure 43) in the virtual environment. They can be represented as a virtual component or a set of virtual components that provide a determined functionality. Once the manufacturing systems are created within the virtual environment, the layout is built by adding the virtual systems to match production requirements.

The engineering phase starts once the functional and operational requirements have been integrated during the concept phase. The engineering phase extends the information contained in the virtual components by adding additional data sets and including design information. The interactions between the systems within the layout are integrated to mirror the real world within the digital simulation. Communication interfaces are defined and developed, considering protocols and signals which will enable the digital twin functionality within the simulation environment provided by VC 4.0. It is during this phase when the connectivity with other ODIN components is enable, exchanging data directly through ad-hoc plug-ins developed on top of the Python and the .Net interface of through OpenFlow achieved through a developed ROS 2 plug-in.

6.3. Development of the ODIN simulation library

Despite each of the pilot cases require dedicated simulation components that will be covered in detail in D5.4 the development of the ODIN simulation library has focused on developing general components that later have been tailored to each of the pilot's use cases requirements. Process Modelling features has been extended and developed new interfaces to extend the capabilities and achieve project requirements. New components have been developed integrating new functionalities and communication interfaces and a new add-on is available.

Process Modelling (PM) is a simulation feature provided with Visual Components 4.0 (available from release 4.2) that allows to manage products, resources, process, tasks and process flows in a layout. This feature has been designed to facilitate to the user of the digital simulation the use of the simulation platform and is in continuous developed. Process Modelling workflow. Figure 45 shows the process modelling workflow.

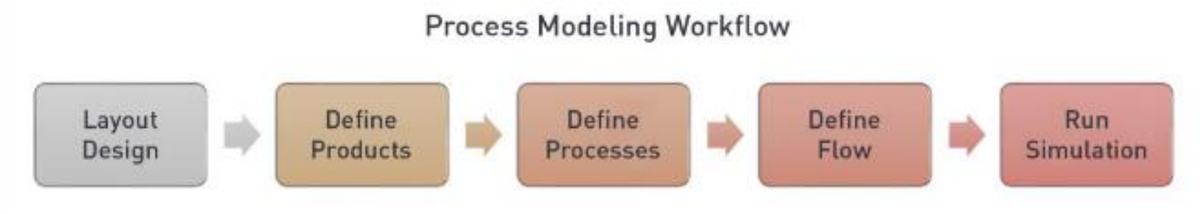

Figure 45: Process Modelling workflow

Products, are defined as the good to be produced, can be a simple part result of a manufacturing process or a complex product result of an assembly process of different parts which can contain different subassemblies. Process is the action or procedure done in a product and can be divided into tasks. Resources are involved in the handling or operation of the components, for processing, can be a conveyor, a robot, a human, a machine. And the Flow specifies the movement of a product from one process to another.

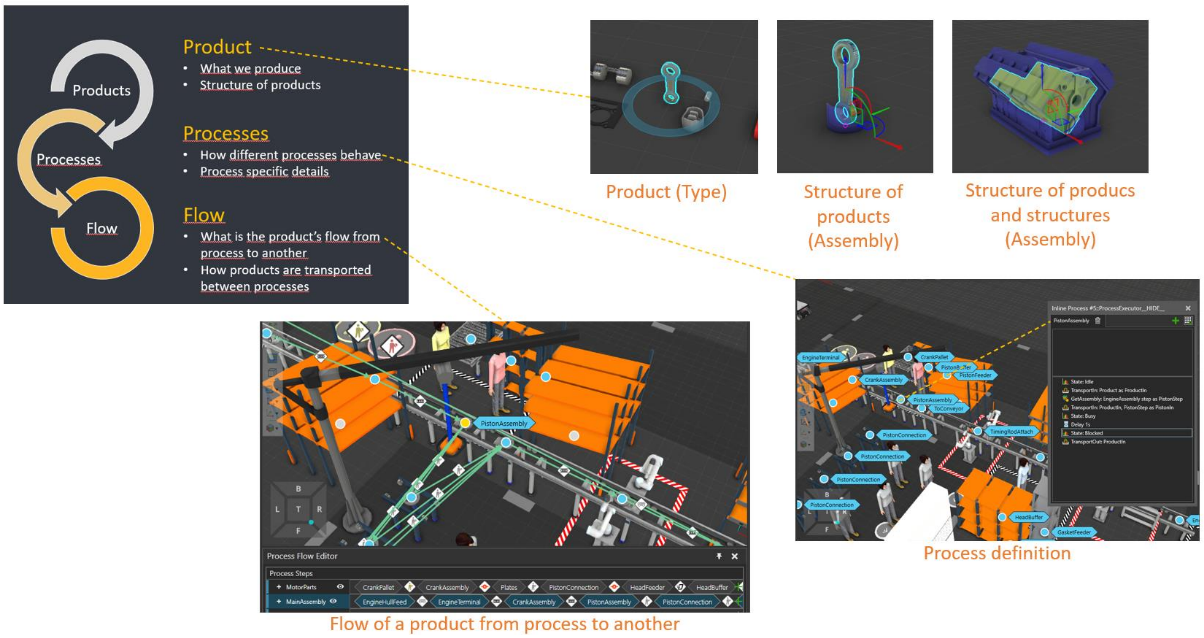

Figure 46: Details of the simplified motor assembly using PM in a HRC cell

The work developed in ODIN for the development of the simulation library have been focused on the extending and develop PM interfaces to extend the capabilities, development of new components, and interfaces and add-on.

6.3.1. Layout components

Layout components includes the components for defining the pilot´s layouts, furthermore these components can be used in combination with resources components to perform actions and processes.

These components have been created adding the behaviours to be integrated with other PM components and extending the interfaces to incorporate with ODIN modules.

6.3.1.1. Trolley with product Mix

Trolley with product Mix (Figure 47), add the capability to the simulation have different kind of products to be assembled in the trolley of the White Goods pilot. Different types of products can be handled in the same simulation components with their references.

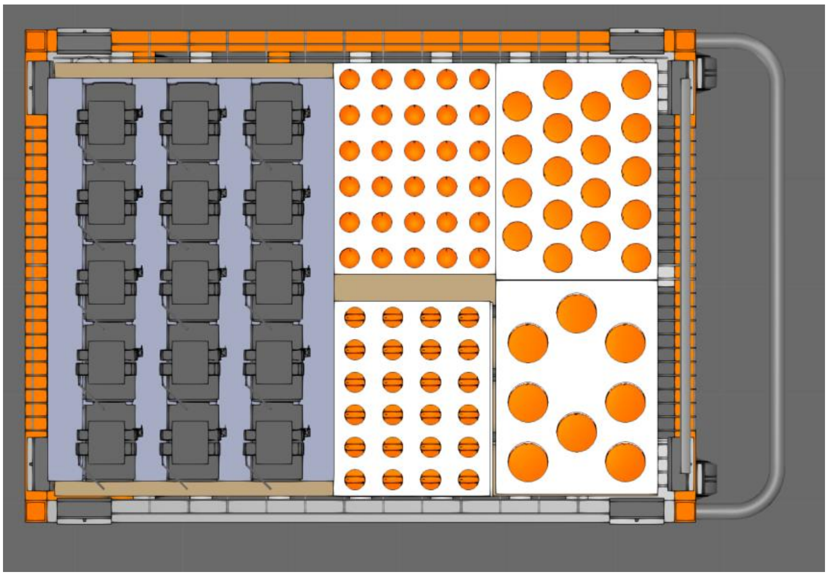

Figure 47: Trolley with product Mix

6.3.1.2. Blisters with cook tops and knobs

Complementing the trolley with product Mix another, the blister with cook tops and knobs have been developed (Figure 48). This component incorporates the parametric properties that allows to easily adapt to new products simply changing the design properties in the simulation UI.

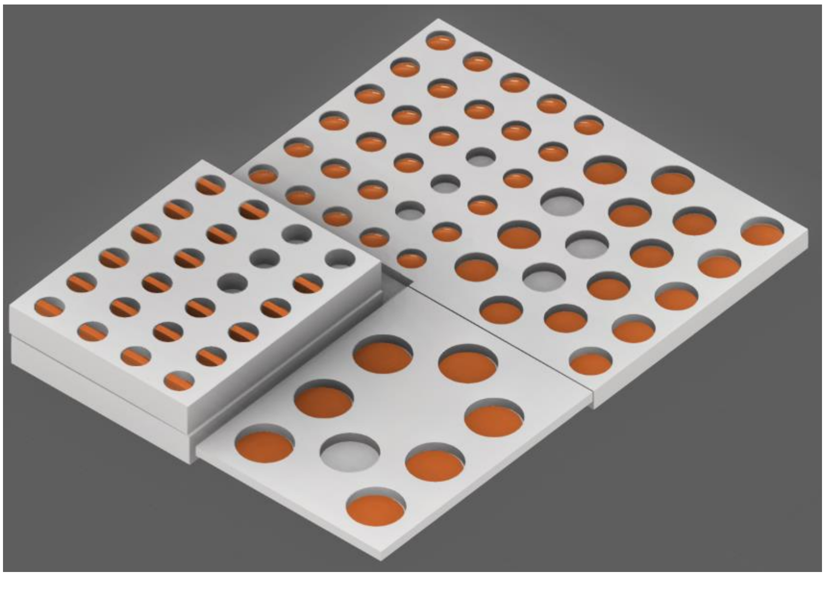

Figure 48: Blisters with cook tops and knobs

6.3.1.3. Vacuum Gripper

The Vacuum Gripper for White Goods pilot (Figure 49) adds the pick and place capability to the component. The simulation component simulates the pick & place cardboard for the product mix trolley.

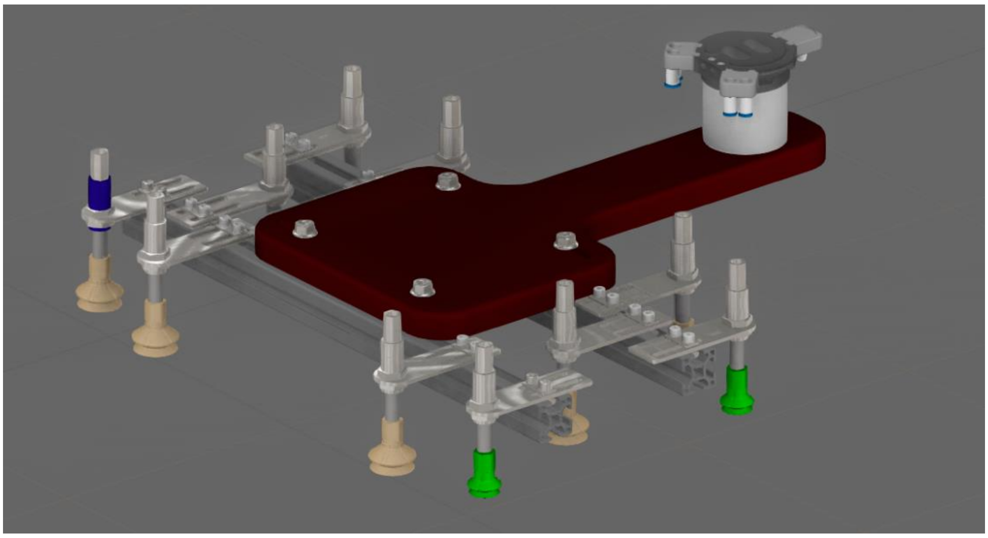

Figure 49: Vacuum Gripper for Whirlpool pilot

6.3.1.4. Magnetic Gripper

Magnetic gripper (Figure 50) for the White Goods pilots incorporates the pick and place capabilities developed. The component mirrors the real one within the digital simulation to pick the transformer, cook top burner cups (small, medium, and large) from the product mix trolley to the shelf.

Figure 50: Magnetic Gripper

6.3.1.5. Flexible Gripper

Flexible Gripper (Figure 51) for the white goods pilot, is used to pick the cooktop knobs from the product mix trolley and place on the shelf.

Figure 51: Flexible Gripper

6.3.2. Product Components

Within ODIN the development within the product has been around improving the assembly process. The assembly in the digital simulation has been improved enhancing the assembly process simulation capability. This process is defined in four phases (Figure 52), product arrives to the station using the transport in (Figure 52 upper left), the product is assembled using the assembly order (Figure 52 upper right), the product assembly is completed (Figure 52 lower left) and the product is leaving the station (Figure 52 lower right).

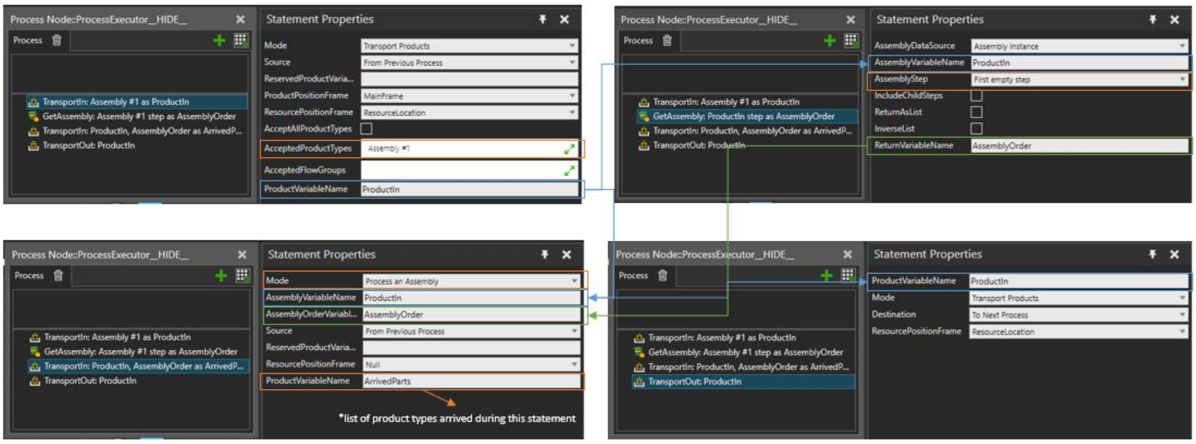

Figure 52: Four phases of the product assembly

In addition to facilitate the creation of assemblies an Assembly add-on has been developed presented in section 6.3.5 of this deliverable.

6.3.3. Resource Components

The human operator has been extended for integrating within ODIN in two directions. First simulation capabilities to integrated with PM has been developed and to integrate with the Digital Twin (section 5.

6.3.3.1. Human Operator

The human operator in the simulation environment is under the resource category. As resource component has been integrated within Process Modelling allowing to create simulations using the same Product-Process-Task-Resource perspective. The human simulation allows walking with or without a product or a tool. The same applies to standing, picking, placing, and working. Animations are defined as robot program routines, and the defaults are used if no custom animations are found for the specific action (Figure 53).

The integration within ODIN can be done generating custom simulations to simulate the operator manufacturing processes, based on the process and product type, the tool component name, or the process name, depending on the active action:

- Processes without a tool or product:
  - <product type name>**_Pick**
  - <product type name>**_Place**
  - <product type name>**_Walk**
  - <product type name>**_Stand**
- Processes with a tool or product:
  - - <tool name>_Walk
  - - <tool name>_Stand
- Work process:
  - Without a tool: <process name>**_Work**
  - With a tool: <tool name>**_Work**

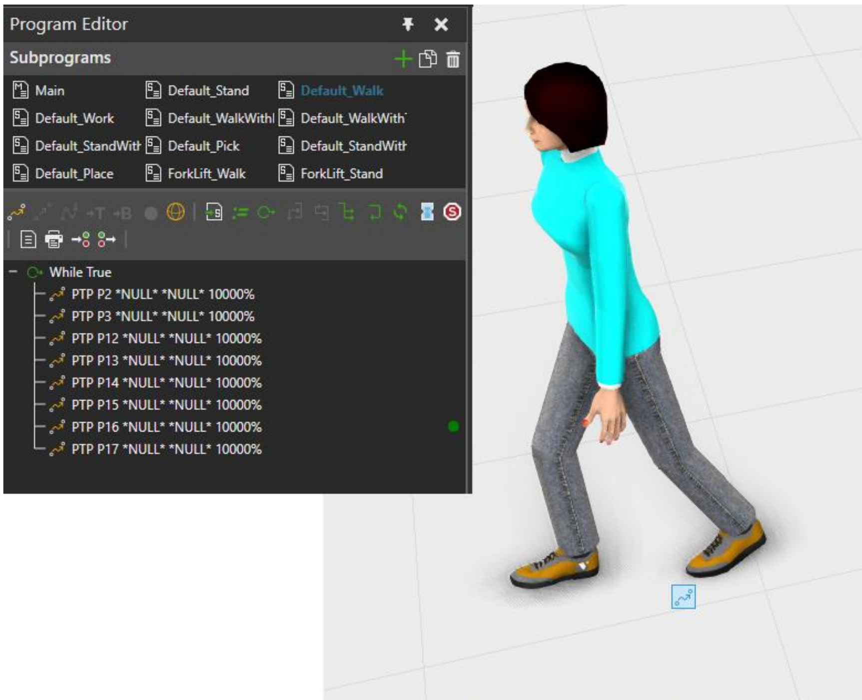

Figure 53: Screenshot of the human operator simulation with the detail of operations

The integration of the human operator with the Digital twin (Section 5) module has required the introduction of the interface for accessing the human joints (Figure 54). The joints in addition to the UI interface are accessible as the rest of the components through the Python and .Net API allowing access to the rest of the ODIN Components.

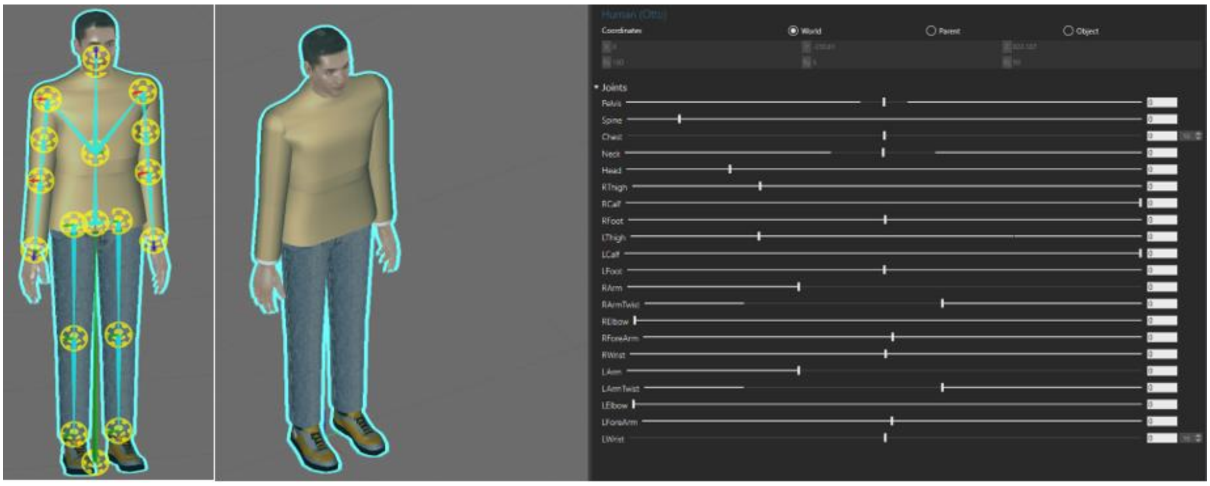

Figure 54: Screenshot of the human operator and UI for interfacing the different body joints

6.3.4. Process Components

6.3.4.1. Missions

This feature has been developed for improving the execution of the task from the resource perspective, which includes: the other of the tasks, optional tasks and customs actions such as move, wait,etc. *Missions* has been developed and integrated with PM to define the task workflow, extending the existing PM existing actions. Their utilization in AGVs simulation (like the ones in the Aerospace and Automotive pilots), allows to define the actual routes and in addition provides additional functionalities like the battery level required for a route and for a full process completion. Missions are granular and one mission can be divided into sub-missions. Additionally, it is included a synchronization functionality to synchronize the resource with other resources. The *mission* simulation feature in addition in AGVs can be also used in human simulation in similar tasks.

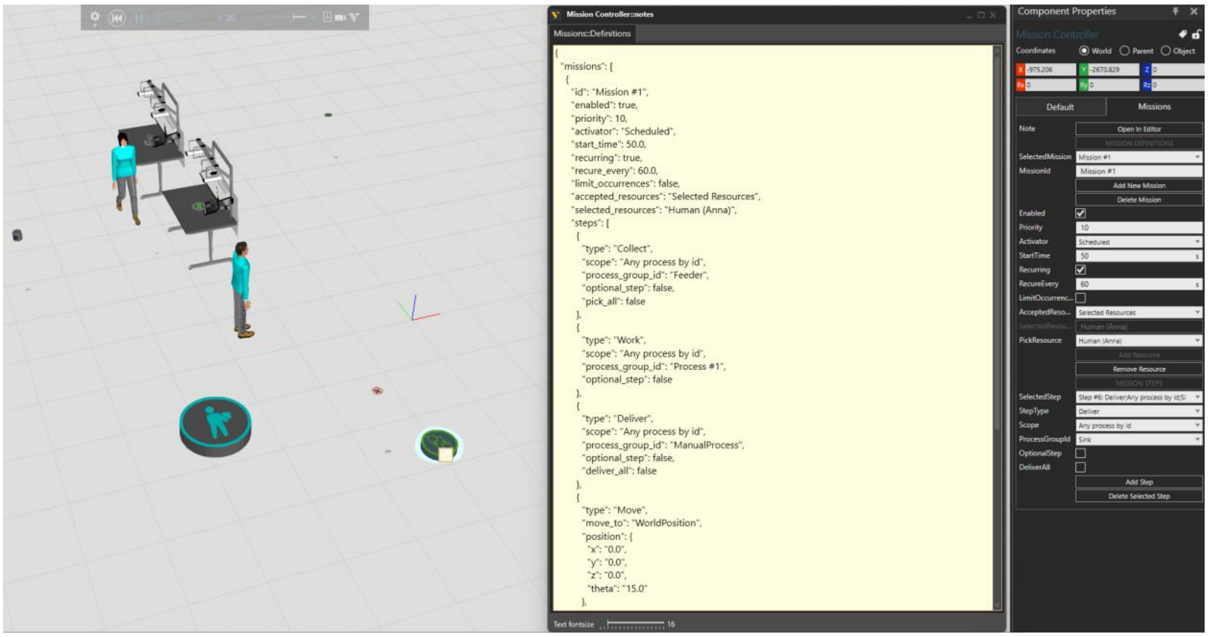

Figure 55: Screenshot of the mission feature for the human operator (left) and the simulation UI configuration tab.

The workflow for creating a *mission* once the simulation layout has been created:

1. The mission controller simulation model is added to the layout
2. The mission controller is connected to the Transport Controller of the resource (AGV, Forklift or Human) (**Figure 56**)
3. The missions are defined in the simulation UI
    a. Missions' parameters (data) are created
        i. Missions can be added, deleted or renamed.
        ii. Missions can be enabled or not
    b. JSON format is used
4. Once the data has been added is possible to run the simulation

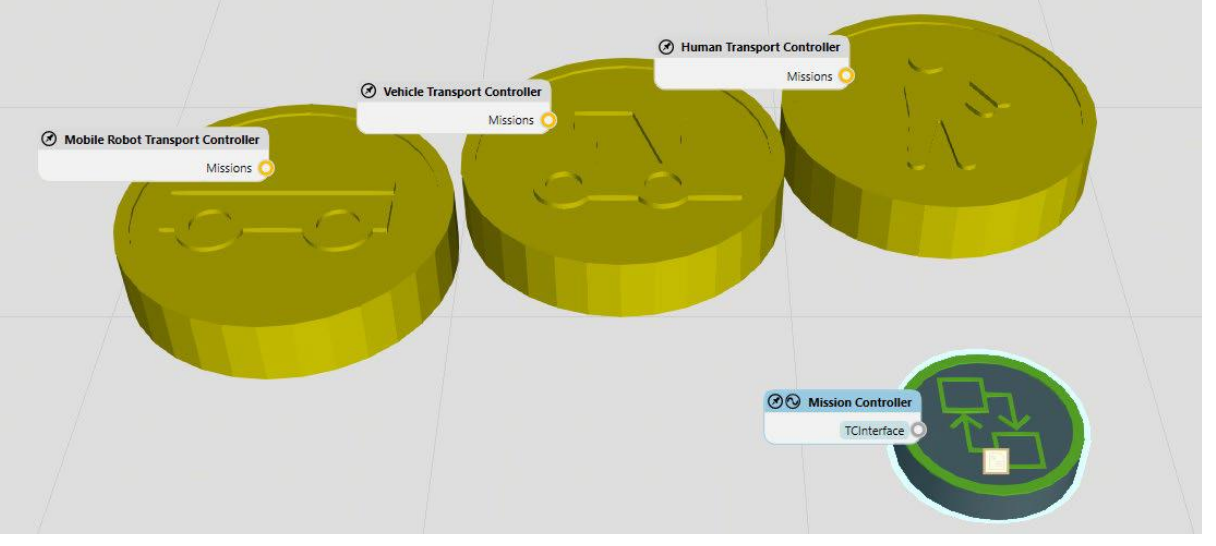

Figure 56: **Detail of the simulation components available for Mission Controller and Transport Controller.**

Missions are accessible through the Python APIs allowing accessing to other ODIN components. The interface allows to define within the missions:

- Mission name
- Move (Component and Position)
- Collect (Transport node, Component, Optional step and Skip immediately)
- Deliver (Transport node, Component, Optional step and Skip immediately)
- Work (Transport node, Component, Optional step and Skip immediately)
- Wait (Wait time)
- Wait Signal (Component, Signal, Wait trigger and time out)
- Send Signal (Component, Signal, and Value)
- Charge (Charging location, Charge limit and Charge until limit)
- Move joint (for actuators operations, Join index, Target value and Motion Time)
- Robot routine (Robot ID, Routine name and Wait execution)

6.3.5. Assembly add-on

The assembly add-on can be used to automatically to create products and assembly types for PM. As source for assembly, allows using hierarchic structure formed by components, nodes, or features. The add-on is accessible, after installation, through the user UI adding a new button under the Process tab.

To use the add-on, a new hierarchic structure should be created using either components, nodes, or features. Select the root component and click *Auto Assembly* button. This will open action panel for the add-on as showed in Figure 57.

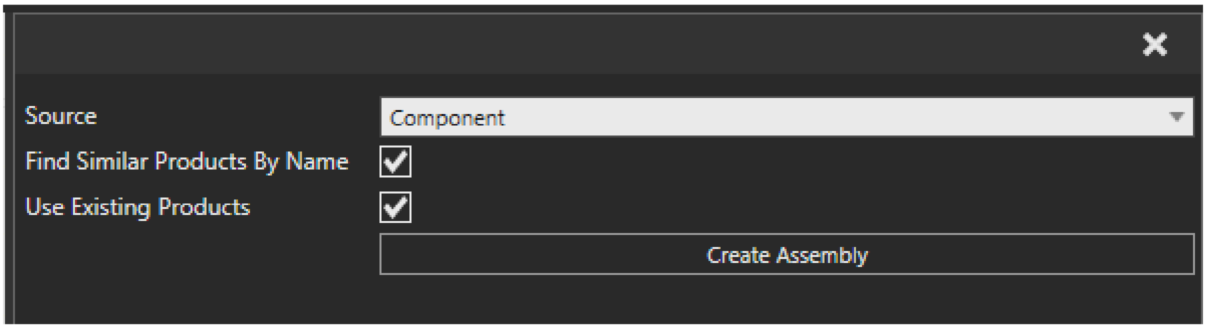

Figure 57: Screenshot of the assembly add-on dialog box

6.4. Integration with other ODIN modules

The digital simulation is integrated with the rest of the ODIN Open components, furthermore it has been integrated with the safety planning design developed by PILZ.

As presented in section 4.3 a prototype tool has been developed for integrating the TAU's matchmaking tool result and resources listed on it to VC 4.0. The tool is designed for production system design, or the reconfiguration phase and it works as plugin in VC 4.0.

The digital representation of the Human Operator available in VC 4.0 can be integrated with the KTH digital twin through the OpenFlow (section 5.4). Figure 39 shows the results of the integration of the pose estimator and the synchronization between the physical operator and the digital operator.

The results of the integration of the AI task planner with VC 4.0 are detailed in section 7.2. The connectivity of the ODIN modules through the OpenFlow module allows the connectivity between the AI task planner and VC 4.0 for layout initialization in the digital simulation and providing the required task plans for execution in VC 4.0. Simulation results can be generated providing evaluation data of the production tasks, which can be supplied from the VC 4.0 to the AI task planner to verify the production plans.

Despite not initially planned, the digital simulation has been extended developing virtual components to support within the digital simulation the safety planning. LMS, PILZ and VIS have been working on the requirements to develop the simulation components required.

6.5. Safety planning using digital simulation

Dedicated simulation components have been developed for safety planning using the digital simulation in cooperation with PILZ as technology provider and LMS for the pilot implementation that will be demonstrated in WP5.

These components facilitate the planning of the safety within the pilot from the initial concepts and have been targeted the safety distance (Figure 58) and the safety zone (Figure 59). Both components have been developed parametric and deploy a UI that allows access to the configuration functionalities. Furthermore, the safety zone includes a detailed UI configuration panel (Figure 60) for detailed configuration of safety zones with positions and max hight for the entire zone.

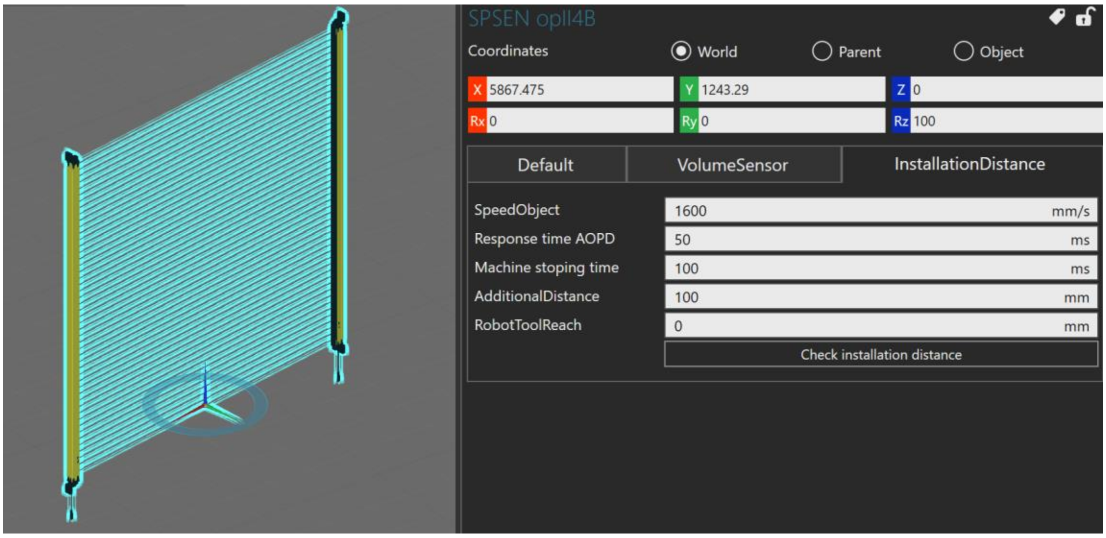

Figure 58: Safety Distance component and UI with access to the functionalities

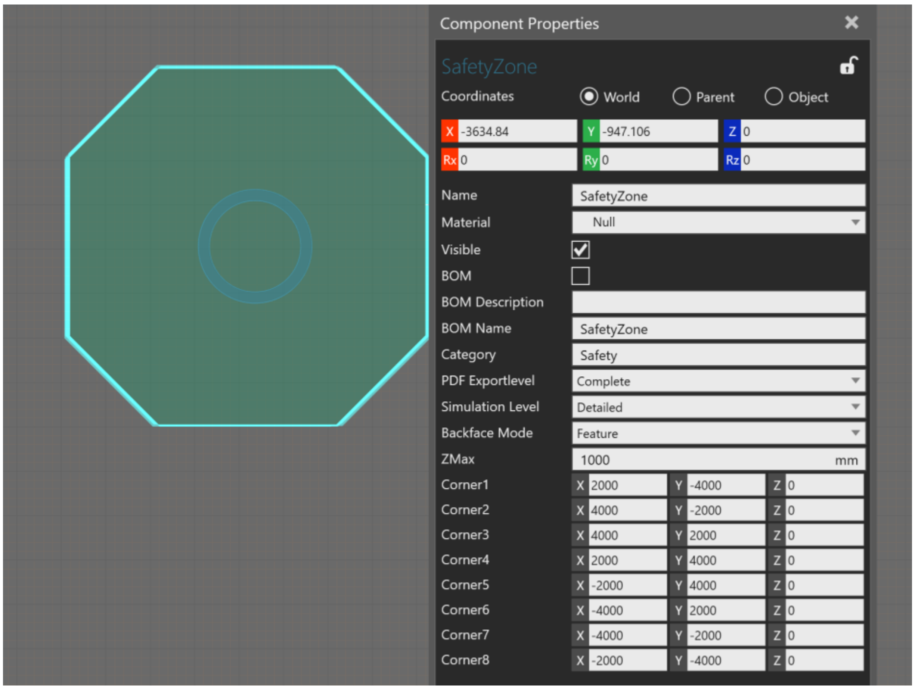

Figure 59: Screenshot of the safety zone with UI with access to the functionalities

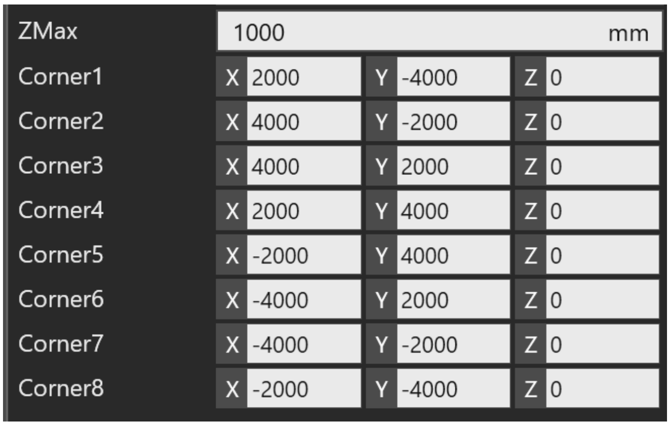

Figure 60: Screenshot of the configuration UI for the safety zone.

6.6. Final implementation

The developments within the digital simulation implemented in VC4.0 are available through the UI provided in VC 4.0 (Figure 61). The virtual components developed are accessible through the electronic catalogue (eCat) where the user can also access to the available public library. The developed assembly add-on is available through a dedicated widget in the UI. In this way the user is able to access the functionalities directly.

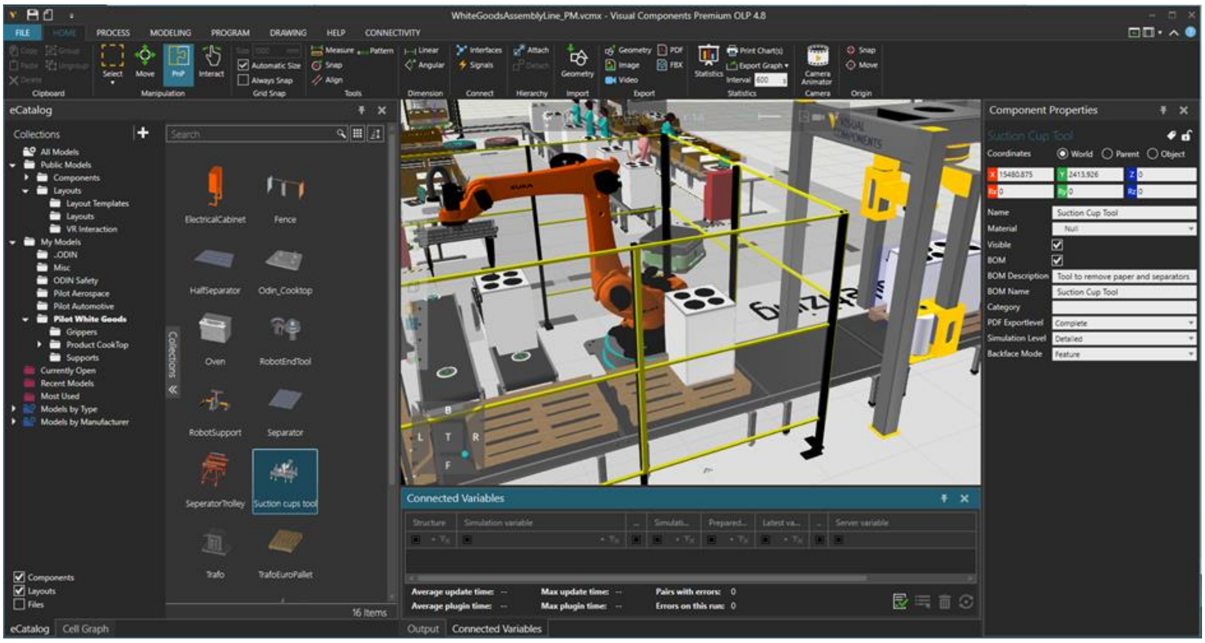

Figure 61: Screenshot of VC 4.0 with the virtual layout view, eCat with access to the developed virtual components and add-on

7. AI TASK PLANNER FOR DECISION MAKING

7.1. Overall methodology

The AI Task Planner of ODIN is the core component which facilitates the cognitive aspects of the line level work re-organization in HRC scenarios. The developed solution is triggered to re-plan and re-allocate the required tasks to the existing resources upon request from the OpenFlow.

The implementation of the task planning approach is based on information of each task available in the database of ODIN. The required feedback about the robotic resources motion planners is used, given that the final goal is to minimize the time required for tasks' execution but also on the allocation of strenuous tasks to robotic resources. Through the implementation of the search-based algorithm for multiple alternative generation presented on D3.1 of ODIN, multi-criteria decision-making mechanisms have been integrated for the evaluation of multiple generator alternatives based on user defined criteria.

The initial version of AI Task Planner's process diagram has been presented in deliverable D3.1 and partners' efforts during the last period of the project were focused on its realization inside ODIN demonstrators. Additionally, new interfaces have been developed for this module's connection with the Digital Twin and the Digital Simulation modules through the OpenFlow for the realization of the re-planning functionality.

The evaluation criteria of ODIN AI task planner have been detailly presented in submitted deliverable D3.1. The workload and resource hierarchical models of ODIN have been finalized (Figure 62 and Figure 63) including information about the robotic resources used under ODIN pilots but also the different tasks required for the investigated assembly operations (Figure 63).

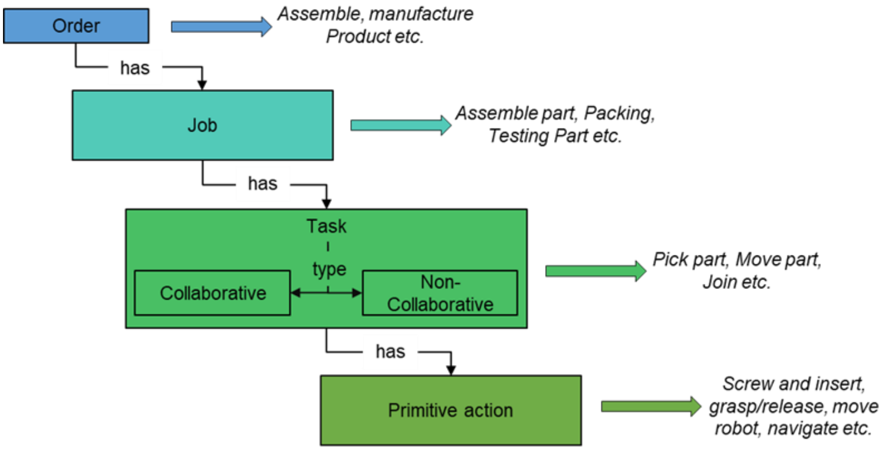

Figure 62: ODIN Workload hierarchical model

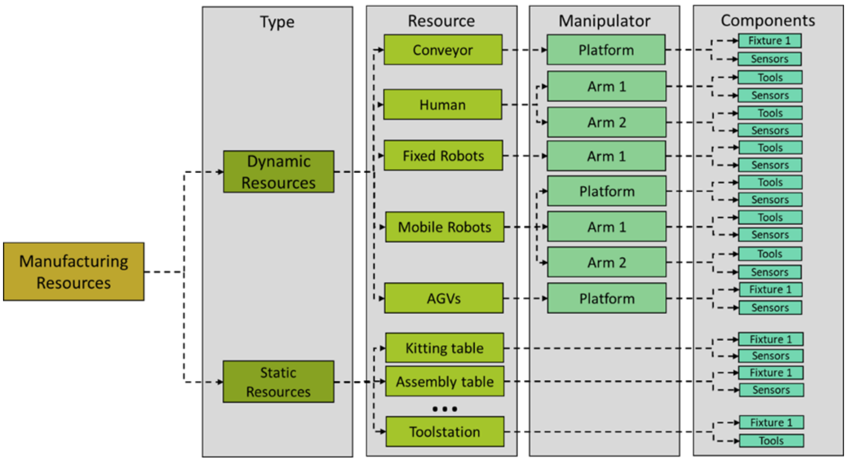

Figure 63: ODIN Resources hierarchical model

As presented in deliverable D3.1, the AI based task planner of ODIN is the key component for the assembly of the investigated products inside ODIN pilots. This module defines the task schedule and each task's assignment to the available resource. In its final form, the AI task planner of ODIN is able to work in two different modes in ODIN pilots:

- *Offline mode*

Running in its offline mode, the AI task planner generates different task plans based on the assembly scenario modelling but also the resource and hierarchical models of the investigated assembly line. The evaluation of the generated task plans is based on the data received from the ODIN's database but also from the simulated execution of the generated task plans by the Digital Simulation module. Inside the Digital Simulation of Visual Components, the required robot motions of modelled assembly tasks are simulated executed calculating the working time for robotic resources.

One important step for the digital simulation of the generated task plans is the initialization of the Digital Simulation layout. The exact replication of the digital layout with the physical one is required for the simulation engine to be able to generate robot motion trajectories similar to the physical ones. For the initialization of the Digital Simulation layout, the AI Task Planner saves data regarding the location of the different VC 4.0 models in a MongoDB database (Figure 64).

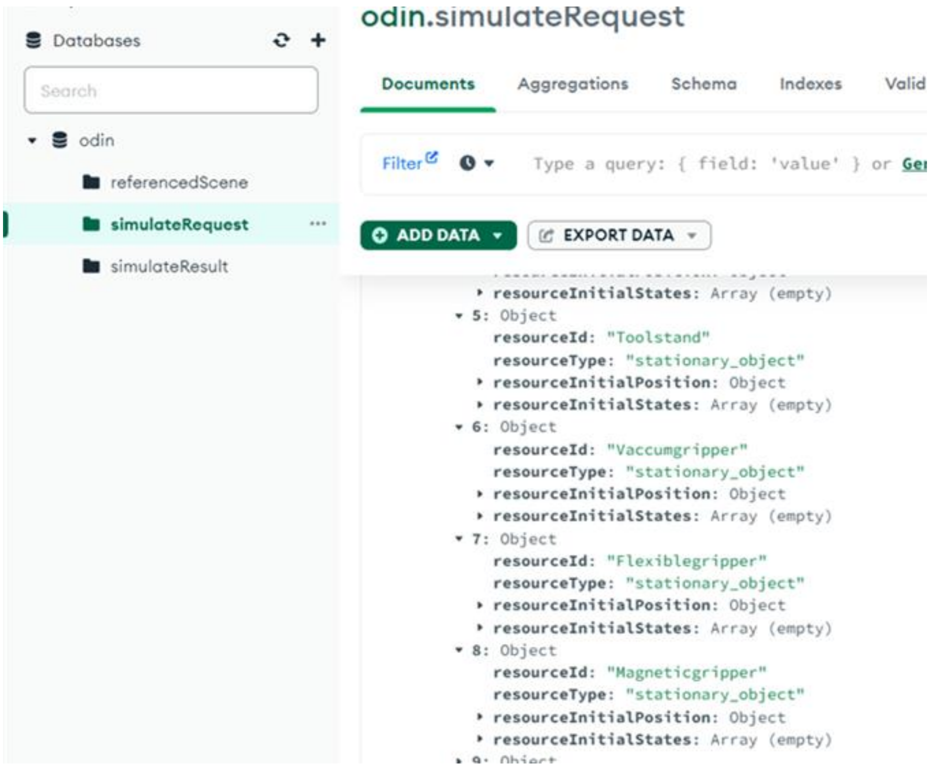

Figure 64: MongoDB database for Simulated layout initialization

This database is accessible by the Digital Simulation in order to load the simulation layout accordingly using the VC 4.0 models available in the VC 4.0 library of the project.

The evaluation data calculated by the Digital Simulation module are passed to the Task Planner module through the OpenFlow as presented to section 7.3. The decision-making engine of AI Task Planner use the evaluation data derived from the Simulation engine but also from the ODIN database and provide the top-rated task plans to the production manager through the developed User Interface (UI) of the Task Planner module.

- *Online mode*

The task planner tool automatically analyses and assess the production tasks in order to rearrange the available production resources/modules in real-time upon request from the OpenFlow. Figure 65 presents the final version of task planner module's methodology scheme including the data flow between different ODIN modules for the online mode of the planner.

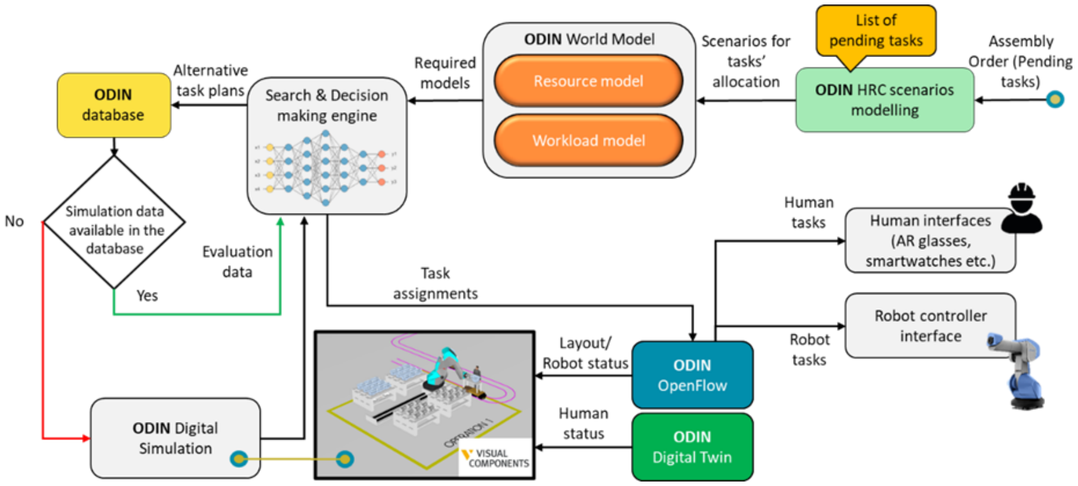

Figure 65: ODIN AI task planner module methodology

The main differences of the AI Task Planner execution when running in Online mode are the following:

- List of re-scheduled tasks

The task plans generated by the AI Task Planner in this case consists of the assembly tasks which have not been executed until the time that the re-scheduling request sent to the planner. The OpenFlow monitors the execution of the assembly tasks and in case of re-planning provides to the AI Task Planner a list of the already executed tasks in order to be excluded during the planning procedure.

- Digital Simulation initialization

In order to calculate the evaluation data in a more accurate way, the initialization of the simulation layout is based on data comes from the Digital Twin and the OpenFlow. The location of the human model in the Visual Components software as well as the initial pose of the robot arm are provided as input for the initialization of the simulation scene to replicate in a more accurate way the physical layout of the work cell as was at the moment the re-planning request sent to the AI Task Planner module.

In online mode, the task planner requires connection with the following ODIN modules (Figure 66):

- OpenFlow
- Digital Simulation
- Digital Twin

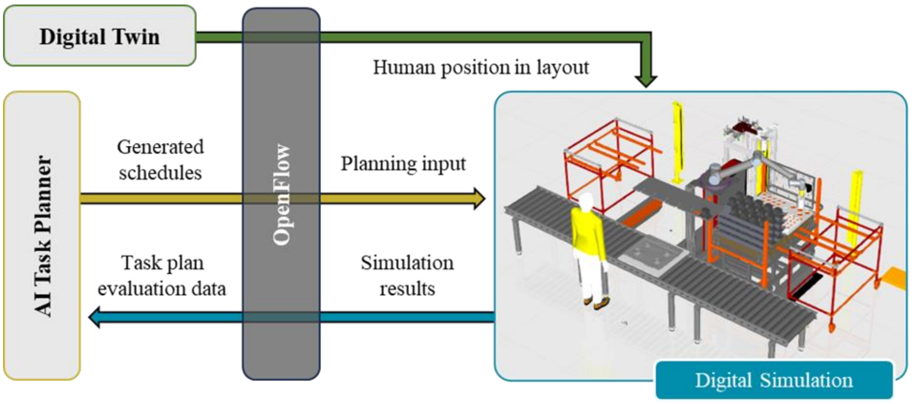

Figure 66: ODIN AI task planner module integration with OpenFlow, Digital Twin and Digital Simulation modules

The integration of the AI Task Planner module with these components is presented in the following section.

7.2. AI Task Planner integration with Digital Components of ODIN

As also presented in submitted deliverable D3.1, ODIN AI Task Planner integration under the industrial pilots of ODIN is based on its integration with OpenFlow module developed under WP5. The initial connectivity of the task planner module with the OpenFlow up to M18 of the project was focused on the implementation and the validation of task planner's offline mode.

In its final deployment, the integration of the AI task planner module with OpenFlow enables the online mode of the planner. The dynamic re-scheduling capabilities of the AI Task Planner are utilized in order to update the production task plan of ODIN pilots upon OpenFlow's request.

As previously presented, in its final form the AI Task Planner is directly connected with the OpenFlow either for providing the required task plans for execution and digital simulation's layout initialization data or for receiving the required evaluation data.

The evaluation of the generated task plans is based on simulated execution of the corresponding task plans from the digital simulation module. The core component for the integration of AI Task Planner module with the simulation engine of Visual Components is the OpenFlow module.

OpenFlow is responsible to provide the required data for layout initialization at the Digital Simulation, by using the information available in the MongoDB database of ODIN and utilizing the VC 4.0 models of the investigated working station. While the AI Task Planner is running in Online mode, the connection between the Digital Twin of the working station and the AI Task Planner is utilized for the accurate initialization of the simulation layout. Using data from the Digital Twin of ODIN, the AI task planner receives information regarding the position of operators inside the HRC workcells. This input is shared by the Task Planner with the Digital Simulation layout for the accurate replication of the physical workstation (Figure 67).

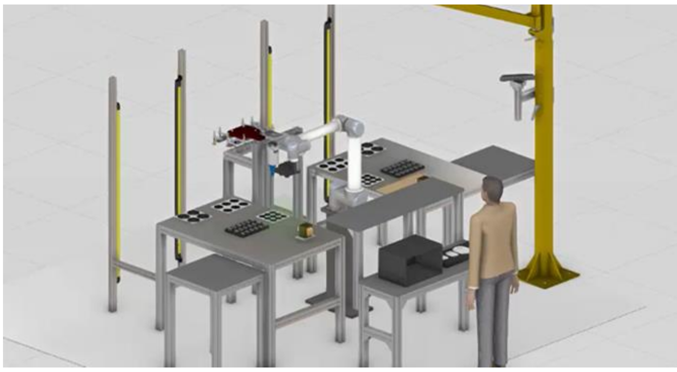

Figure 67: Initialized simulated layout requested by AI Task Planner

After the initialization of the simulation layout, the OpenFlow sends motion trajectory commands to the Digital Simulation through the available ROS2 environment. The calculated evaluation data are visualized through the User Interface of Visual Components 4.0 but are also sent to the OpenFlow through the corresponding ROS2 topic (Figure 68).

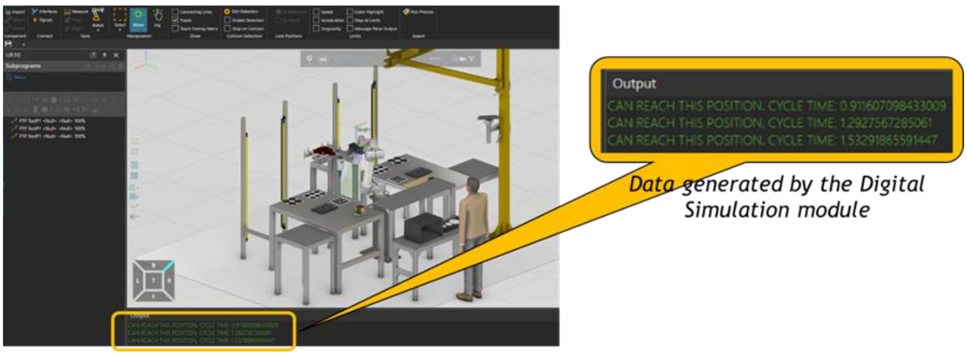

Figure 68: Simulation data used for task plans evaluation

7.3. Final implementation

The first step when using the AI Task Planner is the required login action by the user in order to access the data and the functionalities of the planner module (Figure 69).

Figure 69: AI Task Planner UI – Login page

The initial version of Task Planner User Interface (UI) has been presented in deliverable D3.1. In its final version, the UI consists of several tabs where the production manager can navigate and get access to the different features of the task planner module (Figure 70).

Figure 70: AI Task Planner UI – Home page

Workload information tab

Through the “Workload Information” tab of the UI (Figure 71), the user is able to check the workload and the resource hierarchical models of the investigated pilot cell. These data are stored in the MongoDB database of ODIN and are utilized by the planner. For each task, the user is able to check available data about other tasks that should be executed before this task but also all the tasks that should be executed after this task. Additionally, information about the human ergonomic analysis in case of a task’s execution by a human operator is available through this tab of the UI.

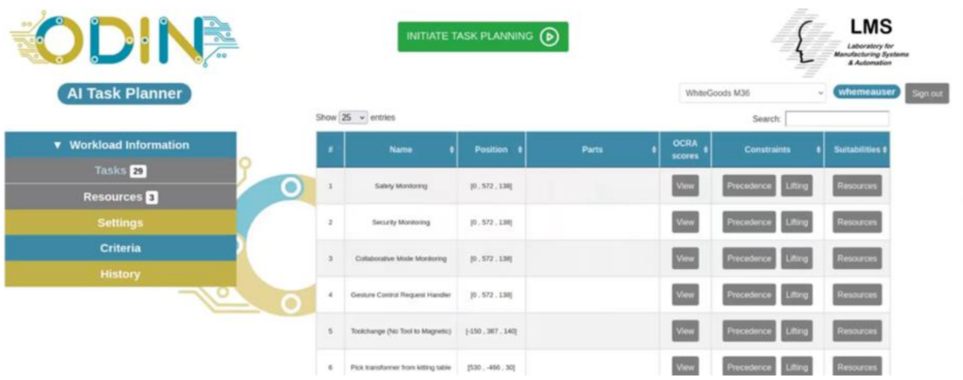

Figure 71: AI Task Planner UI – Workload information tab

Additionally, data related to the resource hierarchical model of the investigated pilot line is available through this tab of the UI. In Figure 72, information about the resource models of the White Goods pilot of ODIN is presented.

Figure 72: AI Task Planner UI – Resource information tab

Settings tab

Inside the "Settings" tab of the UI (Figure 73), the user is able to initialize the evaluation criterias' weight to be used during the evaluation of the generated task plans. These parameters presented also in deliverable D3.1 are the following:

- DH - Decision Horizon: It is the number of that will be examined in each searching step.
- SR - Sampling Rate: This parameter depicts the number of samples that will be defined for each of the decision points of the decision horizon.
- MNA - Maximum Number of Alternatives: This parameter defines the maximum number of alternatives that will be formed at each step of the algorithm. This value is required for the initialization and execution of the Search and Decision making engine of the AI Task Planner for alternative task plans' generation.

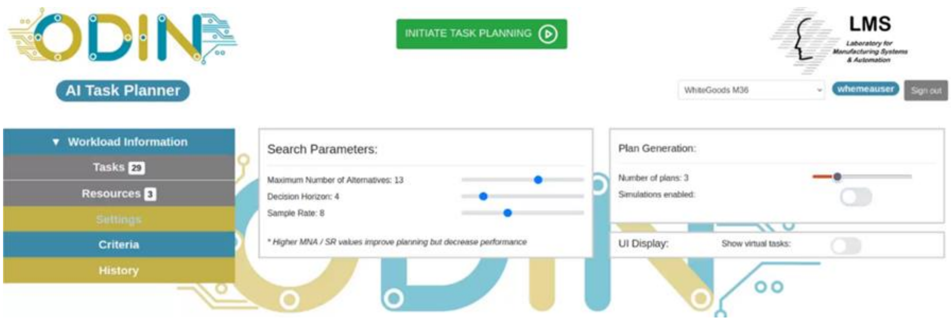

Figure 73: AI Task Planner UI – Settings tab

Criteria tab

Through the "Criteria" tab of the UI (Figure 74), the user of the AI Task Planner module is able to parametrize the weight of each evaluation criteria which are used during the evaluation of the generated task plans by the decision-making algorithm. The evaluation criteria which are used during the evaluation process are the following:

- Flowtime: Total time duration of a generated task plan execution
- Human busy time: Total time duration that a human executes tasks during a generated task plan execution.
- Distance Covered: Distance covered by each resource during a generated task plan execution.
- Ergonomics: How much stressful are the human tasks for the operator based on the generated task plan. OCRA (Occupational Repetitive Action) ergonomics analysis tool is used for the White Goods pilot and RULA (Rapid Upper Limb Assessment) tool for the Automotive pilot.
- Non-Adding values activities time: Time duration that no tasks are executed by any resource.
- Utilization of Resources: Utilization of Resources during a generated task plan execution.

- Safety – HR (Human-Robot) distance: Avoid any plans that do not conform to the minimum distance constraints set between humans and robots.

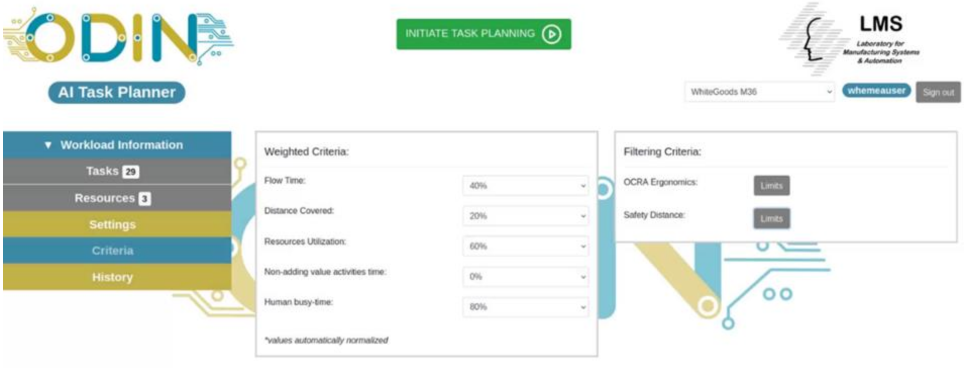

Figure 74. AI Task Planner UI – Criteria tab

The next and the final step for the execution of the AI Task Planner module is its triggering using the "INITIATE TASK PLANNING" button of the UI. After the generation and the evaluation of the different task plans, the user is able to check the requested task plans as presented in Figure 75.

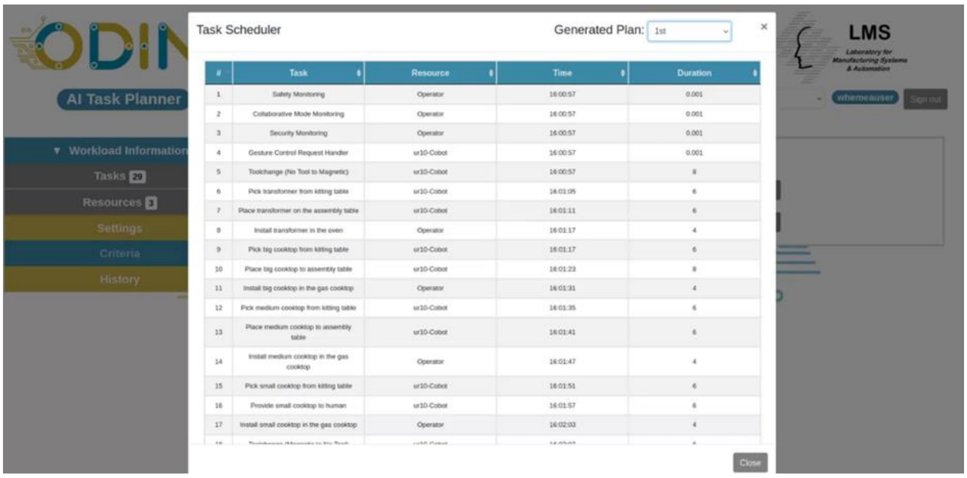

Figure 75: AI Task Planner UI – Execution and validation result

History tab

The results of each Task Planning request are saved in the ODIN database for future usage. The user is able to access this information through the "History tab" (Figure 76) of the developed AI Task Planner module's UI.

Figure 76: AI Task Planner UI – History tab

8. CONCLUSIONS

This deliverable document is focused on presenting the final version of ODIN Digital Component modules and their integration with other modules of the project. In more details, the following modules have been demonstrated in the aforementioned sections of this deliverable:

- Digital description of ODIN Open Components including information about ODIN resources such as size, interfaces, geometry/kinematics, business/design properties, and the specific capabilities.
- Digital Twin of ODIN HRC system, focused on replicating the motions of the human and robot inside the digital environment.
- Digital Simulation for pilot lines simulation execution and validation.
- AI based task planner module for the generation of different task plans and assignments to available resources generation but also their validation using data from the Digital Simulation module.

A communication interface inside the simulation platform for the virtual control and commissioning of ODIN pilots will be presented in D3.4. This communication interface will allow the communication of the DC with the OC to enable virtual control of ODIN pilots in the simulation environment.

Following the initial time plan of the project, the integration of ODIN Digital modules under the pilot lines is presented under D5.4 which demonstrated how the Digital Component provided added value on the pilot cases of the Industrial Component. The final integration and demonstration will take place during the last year of the project and documented under D5.5 on M48.

9. GLOSSARY

AI	Artificial Intelligence
API	Application Program Interface
ARD	Abstract Resource Description
BE	Back end
CAD	Computer Aided Design
DB2	Database 2
DB	Database
DC	Digital Component, ODIN
DH	Decision Horizon
EC	European Commission
EU	European Union
FE	Front end
GUI	Graphical User interface
HR	Human-Robot
HRC	Human Robot Collaboration
HRI	Human Robot Interaction
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IC	Industrial Component, ODIN
ID	Identifier
JSP	Java Server Pages
MNA	Maximum Number of Alternatives
NC	Networked Component, ODIN
OC	Open Component, ODIN
OCRA	Occupational Repetitive Action -method
OISP	On-site Interactive Skill Programming
OWL	Web Ontology Language by W3C
PM	Process Modeling
RCP	Resource Catalogue Platform
RD	Resource Description
RDF	Resource Description Framework
RULA	Rapid Upper Limb Assessment -method
SR	Sampling Rate
SW	Software
UI	User Interface
UKF	Unscented Kalman Filter
URDF	Uniform Resource Description Framework
URL	Uniform Resource Locator
VC 4.0	Visual Components 4.0 (Software)
WP	Work Package
XML	Extensible Markup Language
XSD	XML Schema
XSLT	XML Transformation

10. REFERENCES

1. Anonymous, “Evolvable Ultra-Precision Assembly Systems (EUPASS),” Portal, Cordis EU, 2009. Grant agreement ID: 507978. [Online]. Available: https://cordis.europa.eu/project/id/507978
2. N. Siltala, “Formal Digital Description of Production Equipment Modules for supporting System Design and Deployment,” Tampere University of Technology, 2016. Available: http://urn.fi/URN:ISBN:978-952-15-3783-7
3. Anonymous, “Rapid Reconfiguration of Flexible Production Systems through Capability-based Adaptation, Auto-configuration and Integrated tools for Production Planning (ReCaM),” Portal, Cordis EU, 2018. Grant agreement ID: 680759. [Online]. Available: https://cordis.europa.eu/project/id/680759
4. N. Siltala, “Resource Description, XML Schema.,” 2023. [Online]. Available: https://stage.resourcedescription.rd.tuni.fi/rcp/data/xsd/ResourceDesc_v2-0-1.xsd [Accessed: 22-Dec-2023].
5. E. Järvenpää, “Capability Model, Ontology,” 2023. [Online]. Available: https://resourcedescription.rd.tuni.fi/ontology/capabilityModel.owl . [Accessed: 22-Dec-2023].
6. N. Siltala, “Software Requirements Specification for Resource Catalogue Platform,” Tampere, p. 27, 2022.
7. SmartBear Software, “Swagger. API Development for Everyone.“, 2023. [Online]. Available: https://swagger.io/ [Accessed: 22-Dec-2023].
8. S. Koukas, N. Kousi, S. Aivaliotis, G. Michalos, R. Bröchler, S. Makris, "ODIN architecture enabling reconfigurable human – robot based production lines", Procedia CIRP, Volume 107, pg 1403-1408, (2022)
9. R. Ma, M. Lanz, and N. Siltala, “Visualization concept for representing capability matchmaking results in a virtual environment,” in 31st International Conference on Flexible Automation and Intelligent Manufacturing (FAIM), 2022, p. 9. DOI: 10.1007/978-3-031-17629-6_63
10. Lê Nghi, “Hybrid Database for XML Resource Management,“ 2023. Available: https://urn.fi/URN:NBN:fi:tuni-202304214085
11. Roshan Bist, “Usability Evaluation of Web Application: A Study of Resource Catalogue Platform,” 2024. (In press)
12. Wikipedia, PDCA, available at https://en.wikipedia.org/wiki/PDCA. [Accessed: 22-Dec-2023]