AI·PROFICIENT

Artificial intelligence for improved production efficiency, quality and maintenance

Deliverable 4.6

D4.6: Human-machine interaction, explainable AI and shop-floor feedback

WP 4: Human-machine interfaces, explainable AI and shop-floor feedback

T4.2: Role-specific human-machine interfaces and data visualization

Version: 1.0

Dissemination Level: PU

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

Table of Contents

List of Figures

Acronyms

AI : Artificial intelligence
APK : Android package
AR : Augmented Reality
AS : Auditability system
DCS : Distributed control system
GHO : generative holistic optimizer
HMI : Human machine interface
Iframe : inline frame
IT : Information technology
OCR : Optical character recognition
PEAA : Post-hoc explainable analysis module
QR code : quick-response code
RDF : Resource Description Framework
SDDM : Surrogate data driven model
SPAA : Short-term Post hoc Anomaly Analysis
WP : Work point
XAI : explainable and transparent AI

Disclaimer

This document contains description of the AI-PROFICIENT project work and findings.

The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated in the creation and publication of this document hold any responsibility for actions that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of the AI-PROFICIENT consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 28 Member States of the Union. It is based on the European Communities and the Member States cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors (http://europa.eu/).

AI-PROFICIENT has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.

Title: D4.6: Human-machine interaction, explainable AI and shop-floor feedback

Lead Beneficiary:	TF
Due Date:	31.08.2023.
Submission Date:	31.08.2023.
Status	Final
Description	Human-machine interaction, explainable AI and shop-floor feedback
Authors	TF, IMP, TEK, IBE, ATC
Type	Report
Review Status	~~Draft~~ ~~WP Leader accepted~~ PC + TL accepted
Action Requested	~~To be revised by partners~~ ~~For approval by the WP leader~~ ~~For approval by the Project Coordinator & Technical Leaders~~ For acknowledgement by partners

VERSION	ACTION	OWNER	DATE
0.1	Initial ToC	TF	08.06.2023.
0.2	Sections 1 and 2	TF	30.06.2023
0.3	Major contribution	TF, IMP	28.08.2023
0.4	Ready for review	TF, UL	29.08.2023
1.0	Final version	TF	31.08.2023

Executive Summary

In this document, we provide the summary of activities related to the HMI deployment, XAIs and shop-floor feedback realized across the AI-PROFICIENT pilots. This public document provides an overview of the main achievements related to human-machine interaction systems integration and deployment within WP4. The WP encompasses four tasks, where the successful completion of these tasks has played a crucial role in ensuring effective human-machine interface between the factory managers and shop-operators, the researchers involved in the AI-PROFICIENT platform services and pilots deployment and utilization. This deliverable D4.6 is meant for public dissemination. All the confidential information related to the HMI realization is delivered in a dedicated confidential deliverable D4.2, which is only accessible to the project members, the project officer and the project evaluators.

1 Introduction

This Deliverable delves into a series of intricate scenarios, each showcasing the intricate interplay between technology, user experience, and problem-solving across various domains. There are two main applications that were deployed.

First, an augmented reality AR-enabled mobile application that allows for new user interaction paradigms is introduced. By seamlessly integrating voice-controlled commands with QR code scanning, the application empowers users to engage with their surroundings in a hands-free manner. This synthesis of AR, voice recognition, and real-time data processing exemplifies the transformative potential of immersive technologies in enhancing operational efficiency. Developed for INEOS use case 2 and will be referred to as the *INEOS application* in this deliverable.

Second, a web application displaying information from a sophisticated system designed for real-time AI-driven machinery monitoring and operator feedback. The web application not only presents operators with essential machinery data but also offers AI-generated suggestions for value adjustments. This interactive loop, wherein operators' responses inform AI model refinement, showcases the collaborative synergy between human expertise and algorithmic insights. Developed for the Continental use cases and will be referred to as the *Continental dashboard* in this deliverable.

Throughout the deliverable, the emphasis remains on how technology can be harnessed to empower users, streamline operations, and optimize decision-making processes.

1.1 Scope

The aim of this document is to provide a detailed description of the activities in WP4. All the confidential information related to this activity and deliverable D4.6 is documented in the deliverable D4.2. D4.2 is meant for internal use within the AI-PROFICIENT partners as well as the project officer and experts.

1.2 Audience

The intended audience for this document is for the members of AI-PROFICIENT consortium and Project Officer. Additionally, this document is designed for a wide range of stakeholders, including the European Commission, academics and researchers, reviewers, and the public.

1.3 Relations to other tasks and work packages

Work Package WP4 is initially fed by the technical, functional and ethical requirements coming from WP1. WP5 oversees the architecture and interoperability of tasks T4.2, T 4.3, T4.4 with WP2 and WP3 that deliver the information stream with the AI services and Smart edge components correspondingly. Figure 1 shows the detailed relation of the task T4.2 with other tasks and work packages of the project.

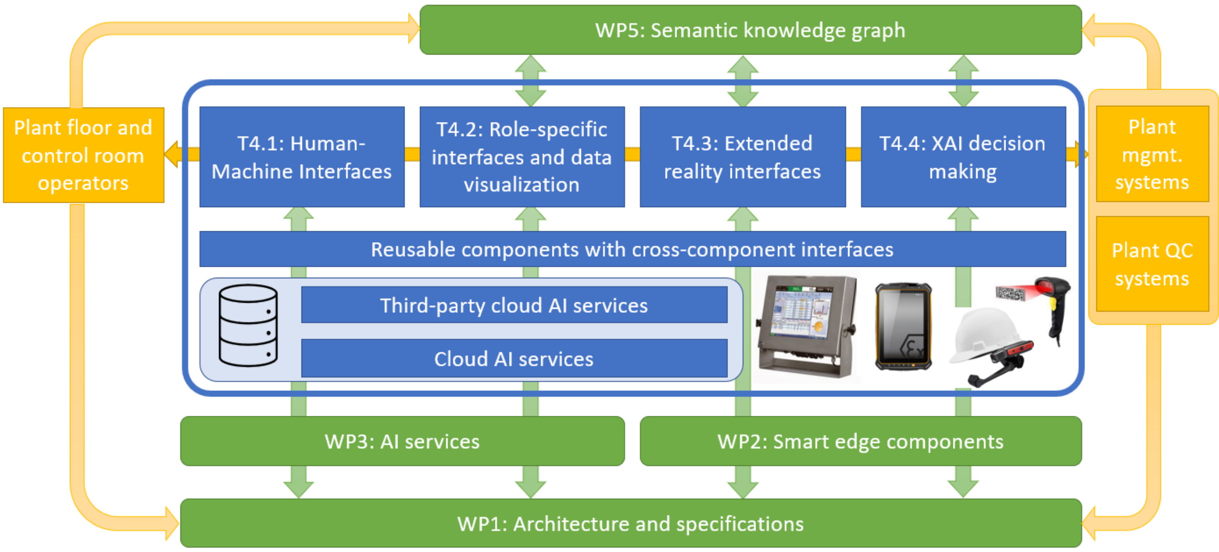

Figure 1: Relation of Task 4.2 with other Tasks and Work Packages

This deliverable D4.6 is based on the initial specifications and requirements detailed in the WP1 deliverables (Figure 2 highlighted in yellow). Further deliverables D5.6 and D5.8 provide the basis for the HMI implementation, whereas the deliverable documents D6.2 and D6.3 impose the functional requirements from the point of view of the end users. Figure 2 summarizes the primary relations of the current deliverable with the other deliverable documents of the AI-PROFICIENT project.

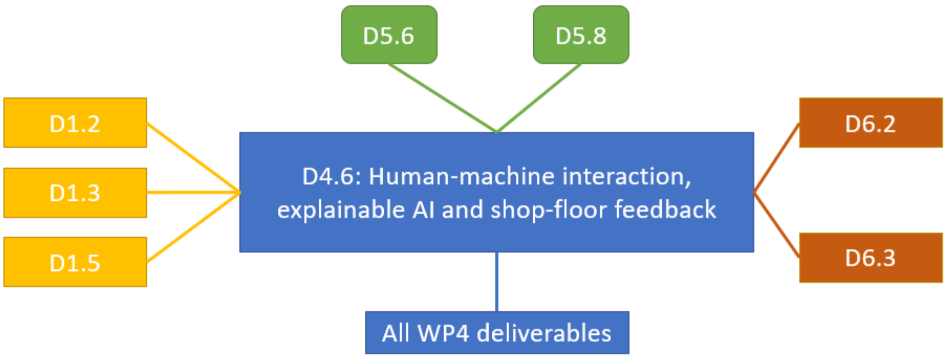

Figure 2: Relation to other deliverables in the project

1.4 Structure

The present document is divided into the following sections:

- **Section 1: Introduction**

  The introduction provides a short introduction of the various topics of this document. The scope and audience are defined.

- **Section 2: Role-specific human-machine interfaces and data visualization**

  This section provides the design process and designs of the HMIs by view and modality. The overall architecture of the platform where the HMI connects to. The interfaces between humans and machines and the interfaces between different components, together with a small summary of the applications' function. The data layer is described, which is where the HMI gets all its data from and converts it to visual information to the operators and plant managers.

Section 2 also goes in depth about the conversational interfaces with the INEOS application and how the human feedback is gathered in the HMI and where it is used. Further, details are given on how the AI system makes suggestions and choices by justifying them.

- **Section 3: Dashboards deployment**

  This section provides details about the deployment of the HMI and how the HMI visually look. A short summary of the function of each HMI is given.

- **Section 4: Stakeholders' access to HMI and its components**

  Section 4 has a table of which stakeholder can access which component.

- **Section 5: Conclusion**

  Section 5 is the conclusion of deliverable 4.2.

- **Section 6: Acknowledgements**

  Section 6 is a list of the acknowledgements.

2 Role-specific human-machine interfaces and data visualization

2.1 HMI concept

The primary goal of the HMI is to provide operators and plant managers with tools to receive and monitor data coming from certain AI-services. The following key concepts are needed to achieve this goal.

- Visualization (large screens and mobile devices)
- User Interaction or feedback
- Real-time data displays
- Alerts and notifications

To accommodate these concepts, a rigorous design process was followed as described in the following subsection.

2.1.1 Design process

Before the design process could begin, it was necessary to receive all the relevant context and information about what should be displayed and to whom. Insights and models are needed from the partners directly. Then, the vision and solution were devised with the corresponding user stories during a multiple-day meeting with the relevant partners. The design process was followed for both the INEOS application and the Continental dashboard. This process is illustrated in Figure 3.

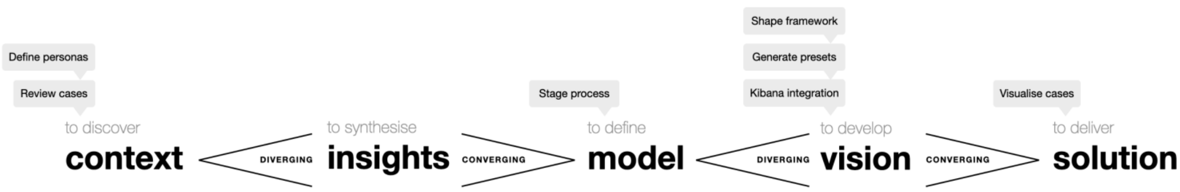

Figure 3: Design process

The Continental dashboard is added to an existing panel of screens within a control room. It should also have the flexibility to be able to be displayed on a mobile device if an operator needs to monitor AI suggestions while on the factory floor. Therefore, the dashboard was designed as a web-application to best adhere to these conditions.

2.1.2 Interaction modalities

The Continental dashboard consists of three main interaction modalities (Figure 4).

- Notify
- Overview
- Report

The notify modality informs operators about certain events or updates from the AI-services. Depending on the needs of the use case, the notification will display different colours. The notification will automatically hide itself after a certain amount of time. This is because the notification is no longer deemed relevant after a certain time period.

The overview modality provides a summary of the information for all the chosen use cases. This can be customized by dragging and dropping the widgets that the user wants to see. Furthermore, Kibana widgets are integrated for a greater view of the data in a visualized form. Kibana, as a mainstream data visualization software for Elasticsearch[1], allows the user to filter and sort data based on their preferences. To drill down, the operator can tap on a widget with the notification to expand the widget and enter the extended view for that widget. This allows the operator to have an overview of the information for that notification.

The report modality allows the plant manager to export the information he wants to pdf format for further analysis or sharing.

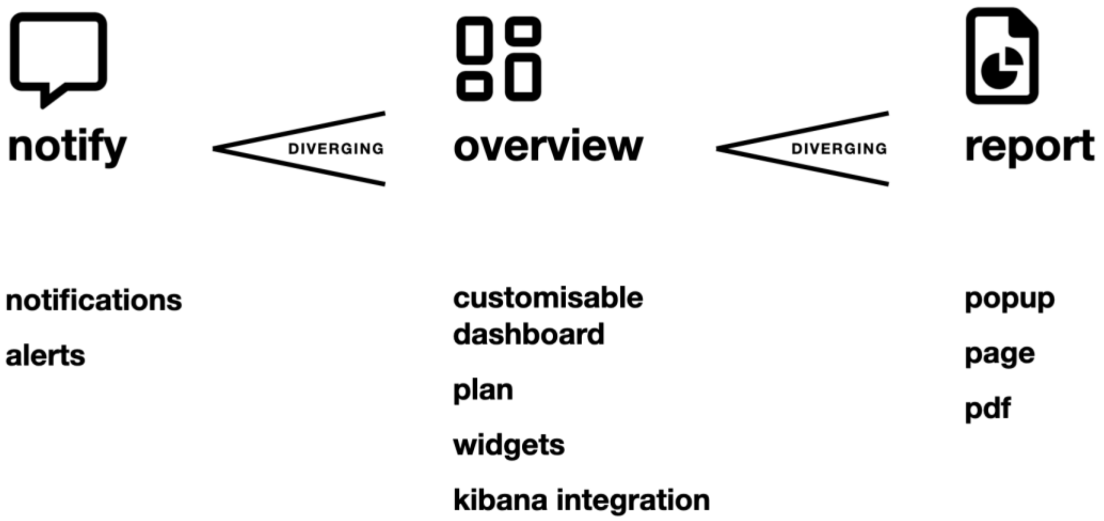

Figure 4: Interaction modalities

2.1.3 Notification View

The notification view is the collection of all the chosen widgets that the user wants to be displayed. It coincides with the overview on Figure 4. Notifications are displayed in each of these widgets. An example of how the notification view looks like can be seen in Figure 5.

[1] https://www.elastic.co/kibana

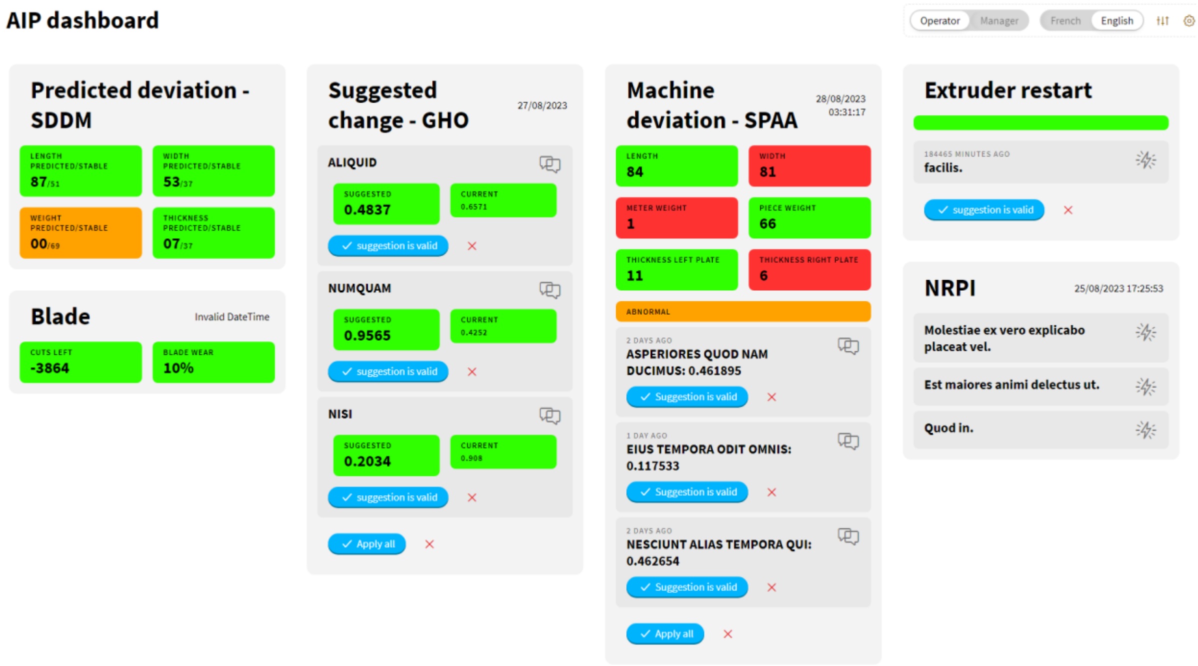

Figure 5: Continental dashboard notification view

2.1.4 Extended View

The extended view is meant for the plant manager in a control room to have a more detailed view beyond the notification view. Here, the Kibana integration displays a visualized form of the information and allows filtering. Figure 6 shows an example of the extended view with substituted data to hide actual sensitive data.

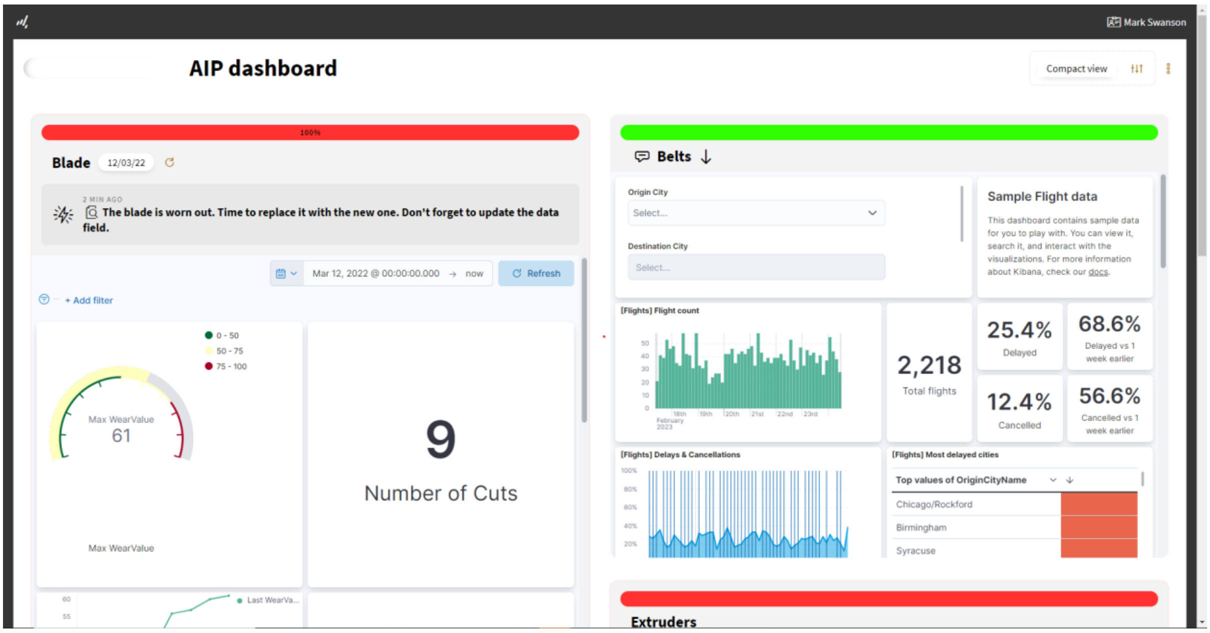

Figure 6: Continental dashboard extended view example

2.1.5 Plant view

The plant view was proposed during the meeting with the partners but not implemented. This view would allow a 3-dimensional display of the plant with tooltips over certain locations which would correspond to the notification widget. Figure 7 displays an example of the proposed plant view design.

Figure 7: Continental dashboard plant view example

2.1.6 INEOS application view

The INEOS app is designed to function primarily on a mobile device. The mobile device is handled while on the factory floor while the operator has safety glasses and gloves on. Because of the limited screen space, user clarity of view and user touch interaction, the application has to have larger text and icons than usual. The amount of information displayed on screen is thus limited. Only the most essential information can be displayed to the user. Figure 21 illustrates the screens of the INEOS application.

2.2 HMI architecture

This section will go into detail about the architecture of the HMI (Human-Machine Interface). This means the Human machine interaction and decision support and platform middleware, as displayed in Figure 8, previously introduced in Deliverable D1.5.

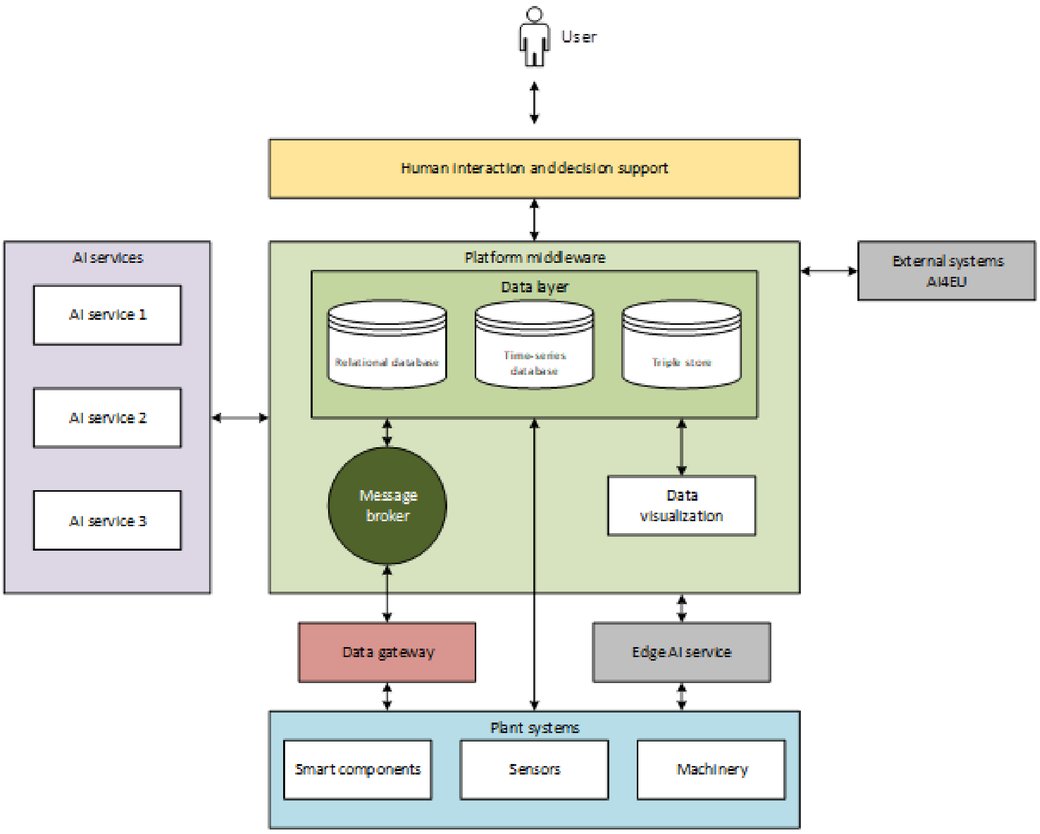

Figure 8: AI-PROFICIENT platform architecture

The HMI is separated into two distinct applications, with different HMI architectures. The Continental dashboard, which is a web application and serves as an overview and notification centre to operators and plant managers. And the INEOS 2 application, which is a mobile application that serves as a tool to streamline the process of linking additives and extruders and receiving feedback of the quality control system in real-time.

The Continental dashboard and all of its components are designed to work in a closed network, ensuring data safety and limiting unauthorized access, as illustrated in Figure 9. The dashboard serves as the Human machine interface between the multiple AI services and the operators and plant managers.

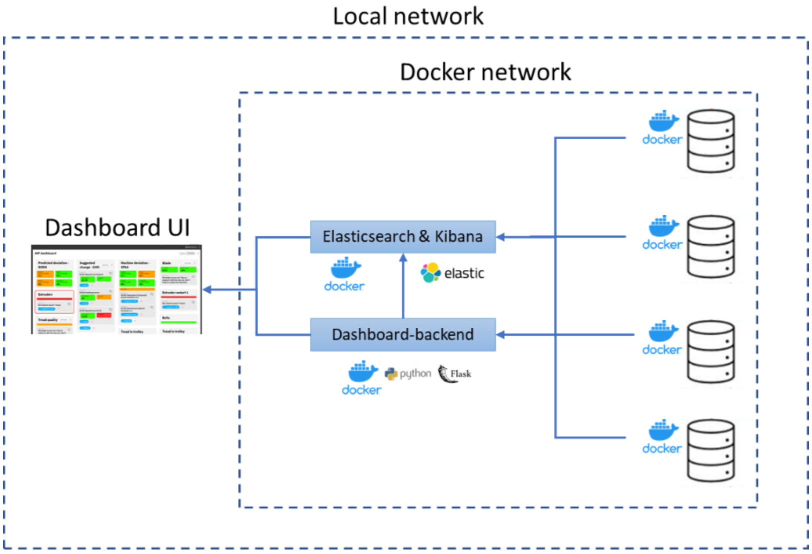

Figure 9: Continental dashboard Network topology

The dashboard receives data from the backend and Kibana and displays the data in compact widgets. It also handles user feedback and passes it to the backend for additional processing. The backend is the system's control centre, overseeing data processing, retrieval, and delivery. It connects to all the databases via a Docker network, that is an OS-level virtualization tool[2]. This simplifies the process of communication between the different containerized services and allows easier scalability. Data is retrieved from the databases, indexed by Elasticsearch, displayed by Kibana. Additional information is sent directly from the backend to the dashboard to be displayed on the widgets.

The INEOS 2 application is designed to operate on a mobile application on the factory floor. Its visual elements are magnified, voice control is used for the head-mounted device. The application communicate with the backend through the internet, which in turn communicates with the DCS (distributed control system) of INEOS. The DCS is within its own controlled network and is only accessible from the backend via an Azure Service Bus, as seen in Figure 10.

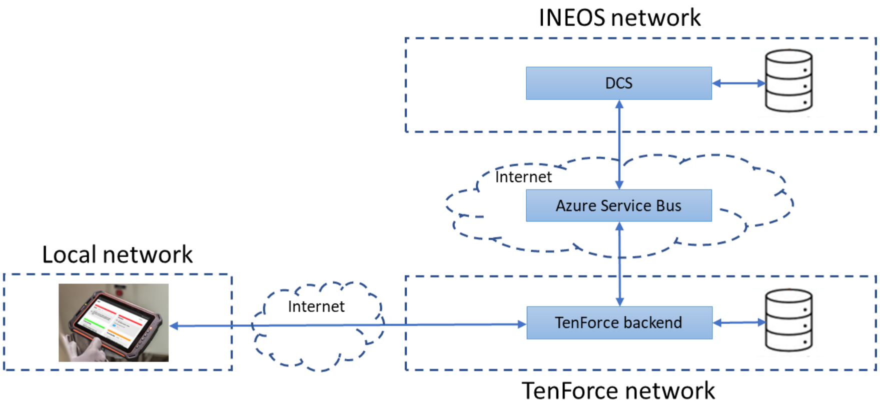

Figure 10: INEOS 2 Network topology

Azure Service Bus provides a secure channel to communicate via HTTPS, and to synchronize the databases with each other. The flow of the Azure Service Bus is illustrated in the diagram below, shown in Figure 11. INEOS' DCS and TenForce's backend can push updates and receive updates

[2] https://www.docker.com/

from each other through Azure service bus's subscription service. The TenForce backend is behind a firewall set to disallow any other traffic from outside the network.

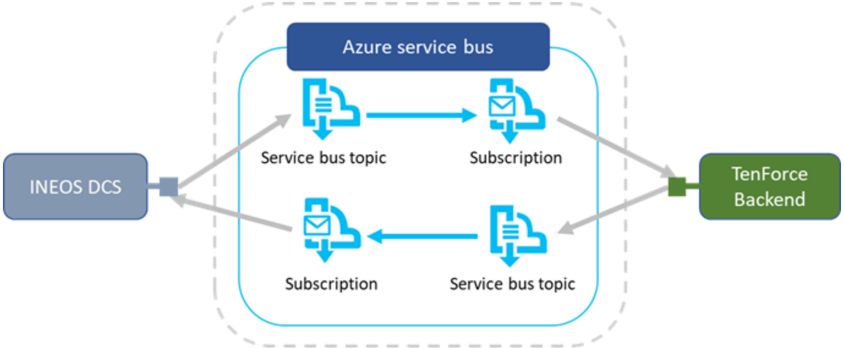

Figure 11: INEOS DCS and TenForce backend connection

Due to being on the factory floor and using wireless internet, there is a possibility of disruption in the connection with the internet. The Azure service bust subscription service can also cause latency. Therefore, the HMI will repeat requests that have failed or were delayed allowing the application's flow to continue after the device reconnects to the internet.

2.2.1 Interfaces

The Continental dashboard is a web application which employs a user-centric interface design, presenting operators and plant managers with a concise overview of the data. Key information is prominently displayed, allowing operators to quickly grasp essential insights. This interface serves as a bridge, enabling users to monitor various systems, processes, or machinery using web browsers on their devices.

The dashboard consists of two views. The notification view and the extended view. The notification view's purpose is to display concise information to the operator in real-time and allowing the operator to provide explicit feedback by accepting or declining the suggesting with a press of a button. The extended view's purpose is to display detailed information to a plant manager to allow insights into certain interactions and historical data.

The information displayed on the dashboard is retrieved from containerized MySQL databases (2.2.2) through a Python backend. AI modules populate these containerized MySQL databases with data, triggering updates to the backend and in turn the frontend. This data is sent to the frontend for the notification view. Simultaneously, data extracted from the MySQL database undergoes transformation within the backend. This includes feature extraction, and integration with other data sources. The backend processes the data into a format suitable for Elasticsearch indexing, optimizing subsequent visualization and analysis. The processed data is visualized with Kibana and displayed in the extended view of the dashboard.

Plant managers are presented with the option to extend their view to access more comprehensive insights. This expandable view offers in-depth details, statistics, and visualizations via Kibana. These visualizations are provided via iframes on the dashboard. Plant managers can interact with Kibana's intuitive interface to explore, filter, and analyse data in real-time. This interface paradigm grants plant managers control over the depth of information they seek.

The INEOS application provides streamlined interactions for the operator with complex systems. It combines image processing, OCR, user interaction, and DCS communication.

The image processing module encompasses image capture, preprocessing, and OCR-based text extraction. Upon image capture, preprocessing techniques optimize image quality for subsequent OCR processing. The OCR service deciphers text from images. The extracted text is then displayed to the user for review.

User interaction is facilitated through an intuitive graphical user interface (GUI). Users are presented with the extracted text and can either confirm it or make modifications. A user-friendly text editing interface empowers users to correct OCR errors or adjust text content. Feedback mechanisms provide real-time validation and aid in minimizing errors during this phase.

Upon user confirmation, the text is communicated to the DCS using a secure communication protocol, via the Azure Service bus. The DCS processes the incoming data and responds with an "okay" or "not okay" status. Error handling mechanisms are embedded to address potential communication failures.

2.2.2 Data layer

The data layer of the AI-PROFICIENT platform is a hybrid collection of data storage and data management technologies fit for purpose. For the sensor readings data technology optimized for time series has been used (InfluxDB), for providing a one-to-one connection with the semantics behind the data an RDF triple store (Virtuoso) has been deployed and finally a common data technology (RDMS) for sharing the results of the AI services has been deployed (MySQL). In the figure below, the time series DB is used to collect and share the plant events, while the RDF store is used to describe the plant configuration information (e.g. the kind of sensor, its location and its purpose). The results of an AI service are stored in a MySQL database.

The deployment technology containerization applied is Docker. Using a containerization approach is beneficial for facilitating a complex collaboration between partners. On the one hand, they can simulate in their local environment the behaviour of their AI services without deploying it on the production environment. At the same time, integration activities when deploying the components on the target environment will not alter the behaviour of the AI service as developed. As such, Docker provides the AI-PROFICIENT platform a consistent runtime environment abstracting the concrete setup of the target deployment machine. More precisely, those detailed requirements of the target deployment machine become visible in this approach, instead of being buried in many distinct configuration steps.

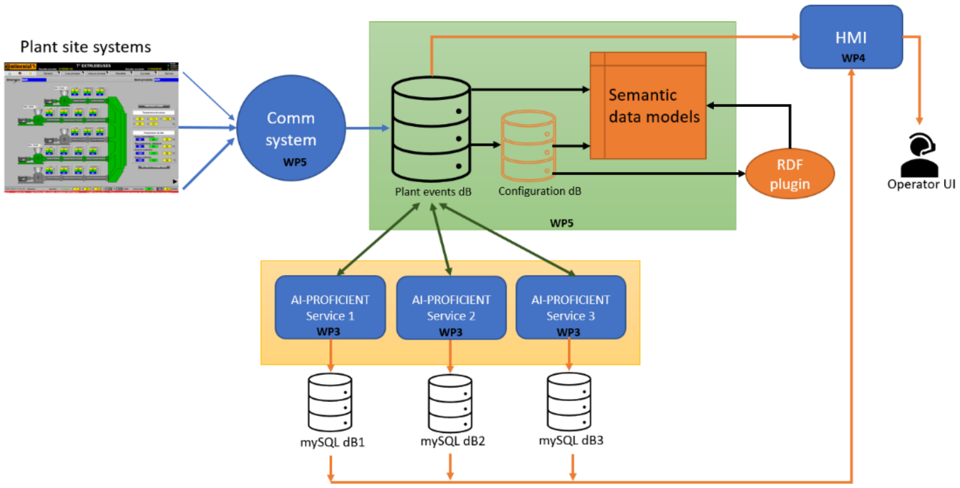

Figure 12: AI-PROFICIENT data layer representation

It has been decided that each AI PROCIFIENT service is responsible for defining and managing its own relational database containing the results it wants to share with other AI services. This follows the concept of microservices (e.g. mysql-service-FCIR-TF), treating the database of each service itself as a separate microservice. Each microservice has its own dedicated database instance, which encapsulates its data and provides specific database-related functionalities which promotes isolated data management, scalability and performance, database specific optimization and deployment flexibility. Extended reality and conversational interfaces for shop floor assistance

This isolation of docker containers ensure that each database-as-a-microservice operates independently and does not interfere with other containers, providing better security by limiting the potential attach surface as well as versioning of container images, making it easier to manage different versions of database instances which can be useful when performing database migrations or rolling back to previous version. Since each microservice has its own docker file, the configuration of each docker container is defined in the .yml file (i.e. docker-compose.yml). The configuration of the docker containers includes the name of the docker image, container name, port number, credentials to access the relational database and volumes where the data from each service is stored.

2.3 Extended reality and conversational interfaces for shop floor assistance

The mobile application for the INEOS 2 use case operates on a head-mounted device, with a voice-controlled implementation. Voice-controlled commands empower operators to interact with the system without using their hands. This interface enhances usability and enables multitasking.

The application flow can be completed through the conversational interface on the head-mounted device. Operators engage with the application through voice commands. The application allows real-time text modification or confirmation. The modification of the text is also handled by the Human feedback mechanisms for AI reinforcement learning. This dynamic interaction loop bridges human intervention and automated data processing.

The application also has an integration of machine learning for adaptive voice recognition (Whisper model by OpenAI), expanding the application to accommodate various industrial-specific languages and settings. Section 2.4 goes more in-depth about the voice interaction system used for human feedback.

2.4 Human feedback mechanisms for AI reinforcement learning

The Continental dashboards are presented by a web application. Human feedback is sent to the backend by clicking certain buttons on the dashboard. All suggestions have this button to send the feedback to the backend. This feedback is sent to the relevant AI-services and stored. The feedback is then used to optimize and refine the existing AI-models, to provide more accurate suggestions.

The INEOS application has a voice interaction feedback mechanism. After a picture is taken and decoded through the OCR system, the results are sent to the operator for verification and possible correction. If the AI module indicates any uncertainty, TEK’s dialogue system takes over (using the app as intermediary between user and dialogue system) and interacts with the user to obtain a correct tag. This dialogue system has been implemented as an API REST service, and consists of three main modules (previously described in deliverable 4.5, and as can be seen in *Figure 16*):

- **Semantic repository (SR)**. This module is the core of the dialogue system, as both the dialogue flow and the domain knowledge (in this case, the different tags that can be identified) depend on its modelling (by using the Task-Oriented Dialogue system Ontology (TODO)) and instantiation. The RDF store Virtuoso has been used to store this knowledge and to make it available through a SPARQL endpoint.
- **Dialogue manager (DM)**. This module is in charge of, considering the outcomes and knowledge retrieved from the SR, manages the dialogue process to obtain, in the end, a correct tag.
- **Polarity interpreter (PI)**. Given certain dialogue situations in which the user needs to confirm an information, this module is in charge of determining whether a sequence has a positive value (e.g., *yes*, *that's it*) or negative value (e.g., *no*, *not at all*).

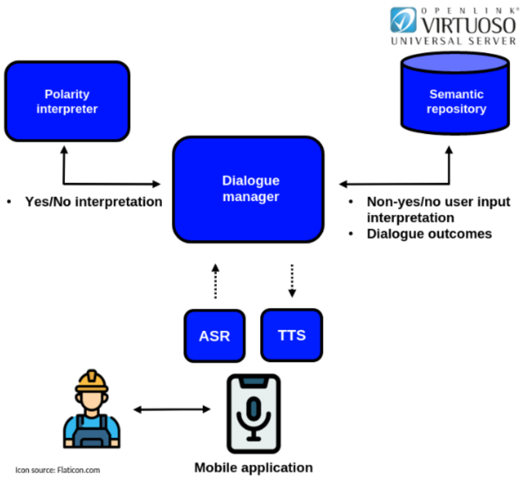

Figure 13: Dialogue system architecture

When the user provides the tag through voice, and the app gets the corresponding transcription using ATC's Speech to Text (S2T) module, the DM uses that input to retrieve the closest tag(s) in the SR. Given the SR output, three possible situations may arise, with different associated outcomes:

a) Multiple options can correspond with the information by the user (e.g., the user says ABC and the system returns the options ABC1 and ABC2): the system will present, one by one, the different options to the user until they confirm it is the intended tag.
b) Only one option is obtained: the dialogue system will return that tag.
c) No options are available: the dialogue system will ask the user to repeat the tag and the dialogue process is restarted re-trying to get a correct tag.

The operator can't always easily type on the tablet because they are wearing protective gear, or they might be using a hands-free AR device, thus a Speech to Text module was implemented as a REST API to aid the operator in making changes on the mobile application without the need to have the device in hand. The mobile application sends audio clips in intervals to ATC's Speech to Text module. This allows the operator to manually make changes in the mobile application with only speech and the aid of the previously described dialogue system.

All in all, the operator input (either obtained through OCR or dialogue) is sent to the TenForce backend and forwarded to the INEOS DCS system for verification. The DCS's response is shown to the operator, and all updated data is stored for reinforcement learning.

Any manual changes made by the operator is stored, so they can be used at a later date to refine either the OCR model or the label matching model.

2.5 Explainable and transparent AI decision making

One of the crucial parts of WP4 is involving explainable and transparent AI (XAI) decision making in the recommendation system. XAI, a subset of artificial intelligence (AI), centers on creating systems capable of offering justifications for their actions and choices. This encompasses revealing both the inner mechanisms of the AI system and the logic underpinning its decisions. These systems aim to be clear and understandable, ultimately bolstering trust in AI and enhancing user satisfaction. By elucidating the rationales behind AI's decisions, users can gain deeper insights into its operations and the reasons behind specific choices, thereby fostering a sense of reliance and certainty in the system.
Within AI-PROFICIENT project, XAI has been studied within T4.4 and reported within D4.4 AI-PROFICIENT approach for XAI. However, the main outcomes are as follows:

- Surrogate data driven model (SDDM) service is XAI based model supporting generative holistic optimizer (GHO) for the purpose of forecasting product characteristics based on the current process parameters. This service is based on different machine learning models which are used for characteristics estimation and are wrapped by XAI models, which are providing the most probable causes of a particular estimation. For example, if it was forecasted that weight will be out of bounds, the most probable process characteristic for this degradation will also be provided. This service has been developed mainly for CONTI10 use case and is also present within CONTI2. The outputs of the aforementioned SDDM are presented to the operator in real-time through dashboard explained in section 2.6.2.

- Post-hoc explainable analysis module (PEAA) service is XAI based model developed for the purpose of analysis of the historical data and the most probable causes of the previous degradation. PEAA is developed as a part of CONTI10 use case and is presented though the corresponding dashboard for the process manager.
- Auditability system (AS) is XAI based service which, contrary to the previous two, was not developed to support plant employees, but rather developers of AI services for the plants. It is based on the semantic technologies modeling development system information, AI model implementation, AI model metadata, etc. This service was tested for CONTI2 and CONTI5 use cases.

3 Dashboards deployment

The dashboard uses widgets to provide information for each use case to the relevant operators. The operators can choose which widgets they would like to have displayed by dragging and dropping them in the list displayed in Figure 14. The dashboard is deployed on the continental server and can be accessed by all authorized personnel to that server. Some widgets also provide an extended view.

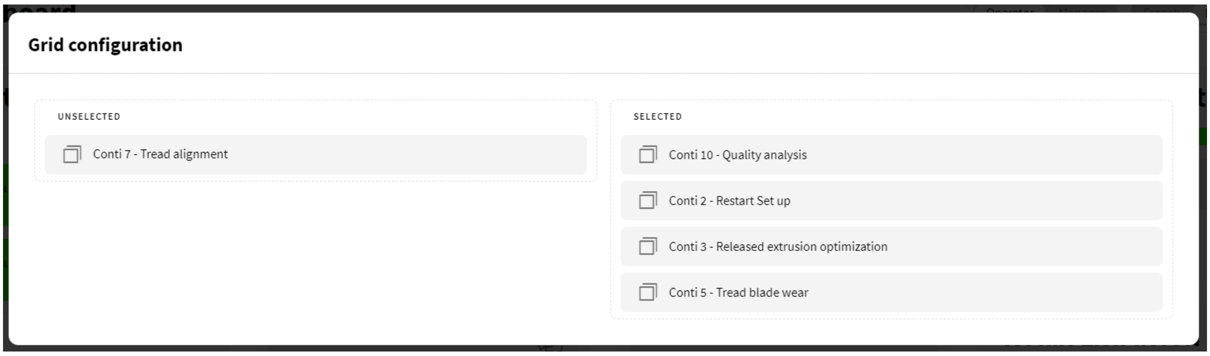

Figure 14: Widget choice list

The following subsections provide a view of the deployed applications per use case, with a short summary for context. The figures have been slightly edited to hide sensitive data, while others use substitute data.

3.1.1 CONTI2 and CONTI-5 deployment

The widget of Conti2 displays the state of the extruder when the production is (re)started. The state and suggestions are provided by the AI-service. An operator can indicate that the suggestion, is valid or decline the suggestion which is then stored in a separate database. Figure 15 is the widget on the deployed dashboard.

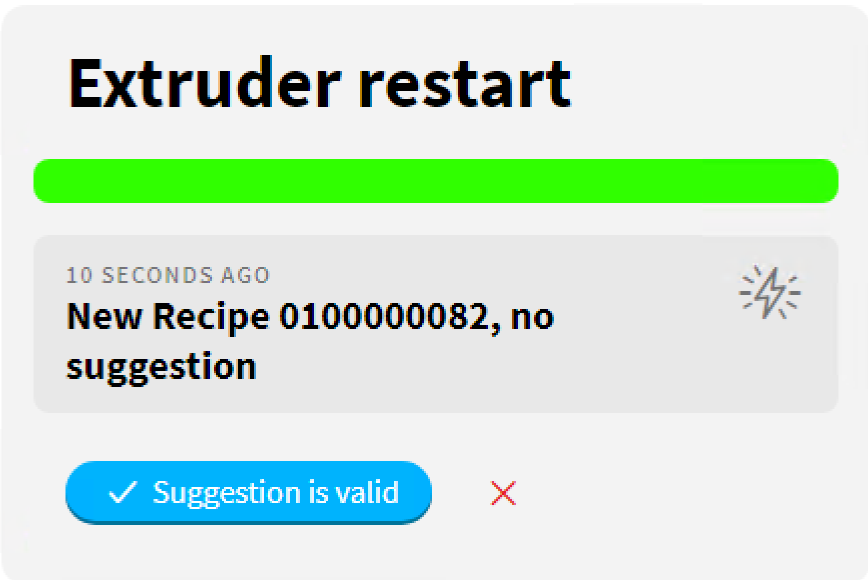

Figure 15: Conti2 widget

Conti5's widget allows an operator to have an eye on how dull a blade is. The dullness of the blade is displayed as the blade wear. The widget as seen in the deployment is illustrated in Figure 16. The following are displayed to the operator:

- The number of cuts left until the blade wear value reached 100%
- The blade wear – a percentage value of how dull the blade is.
- The date when the blade was last replaced.

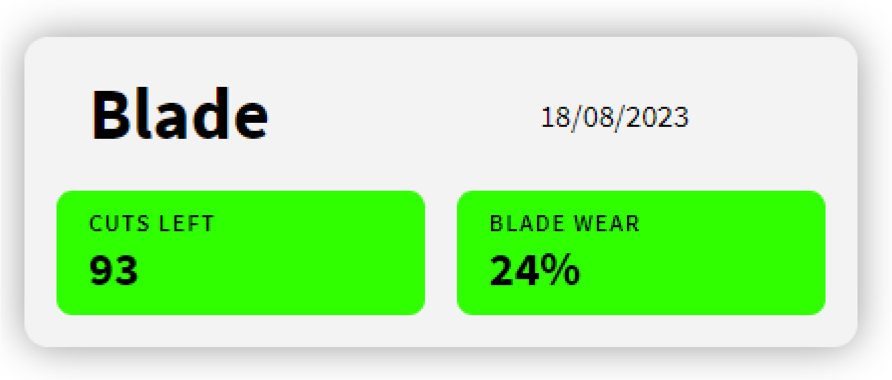

Figure 16: Conti5 widget

3.1.2 CONTI10 deployment

Conti10 is split into 3 widgets; Short-term Post hoc Anomaly Analysis (SPAA), SDDM and GHO.

3.1.2.1 SPAA

SPAA tracks the values of the products and the parameters of the machine setpoints. Whenever an anomaly occurs, it finds the root cause of the anomaly and displays it with the option for the operator to validate the suggestion. The deployed widget with edited values is seen in Figure 17. Below is the list of attributes shown:

- The current attributes of the product versus the setpoint (stable).
- The status of the machine parameters that the AI service monitors.
- A list of machine parameters that have an influence on the abnormal state.

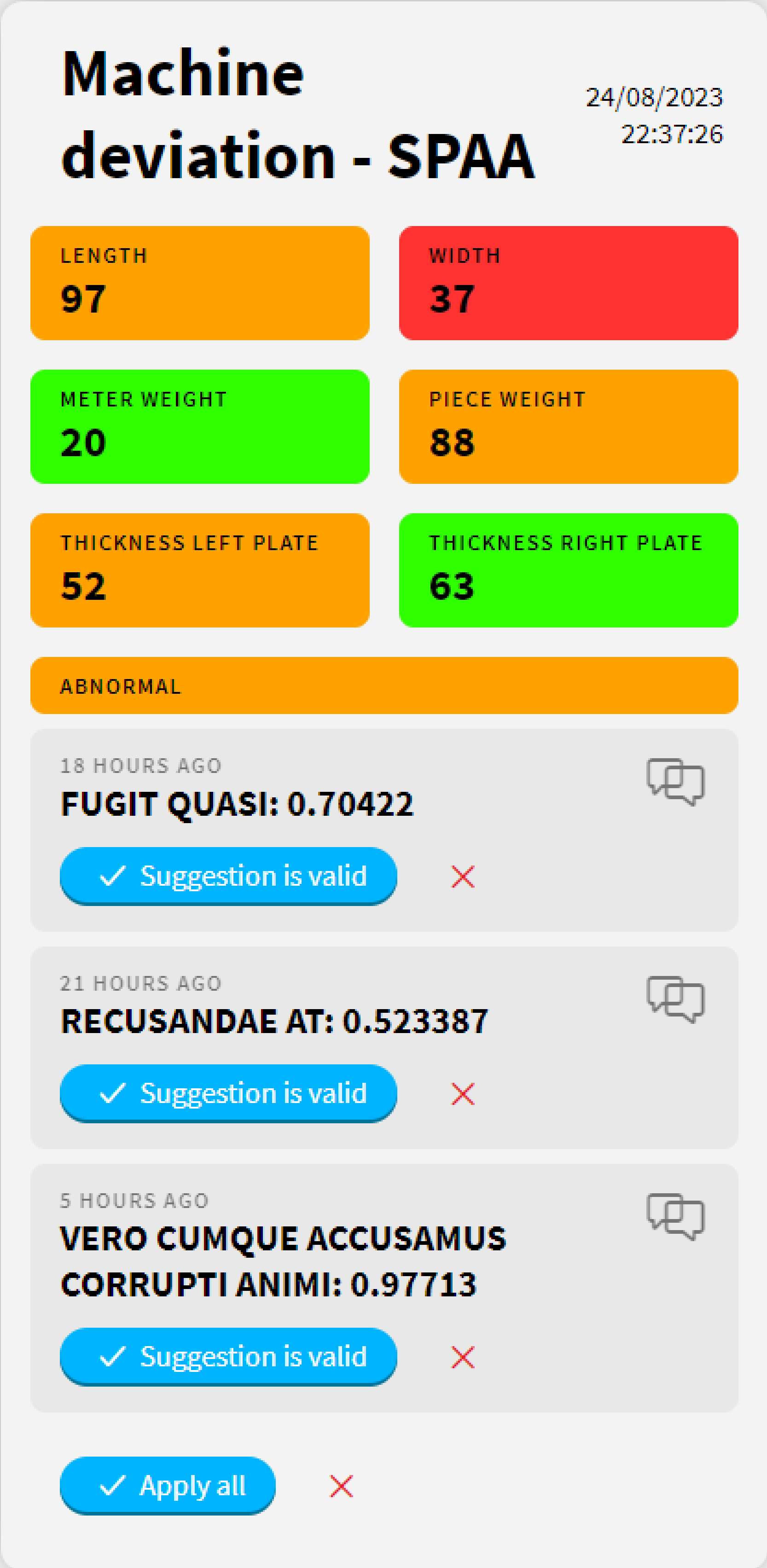

Figure 17: Conti10 widget SPAA

3.1.2.2 SDDM

SDDM displayed the predicted values of the product and can be seen in Figure 18.

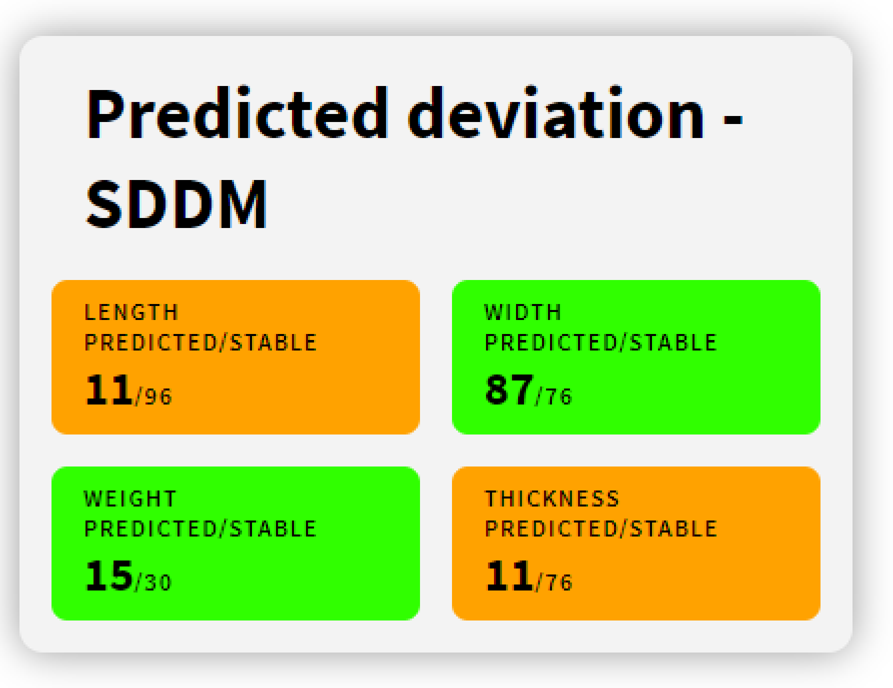

Figure 18: Conti10 widget SDDM

3.1.2.3 GHO

The GHO widget displays the suggestions to arrive to an optimal product state. SDDM or SPAA can trigger suggestions. SDDM triggers suggestions by predicting possible deviations that will happen based on the current trends. This will cause the GHO to provide suggestions to rectify the deviation before it happens. A maximum of three suggestions are displayed to the operator. The visualization is seen in Figure 19.

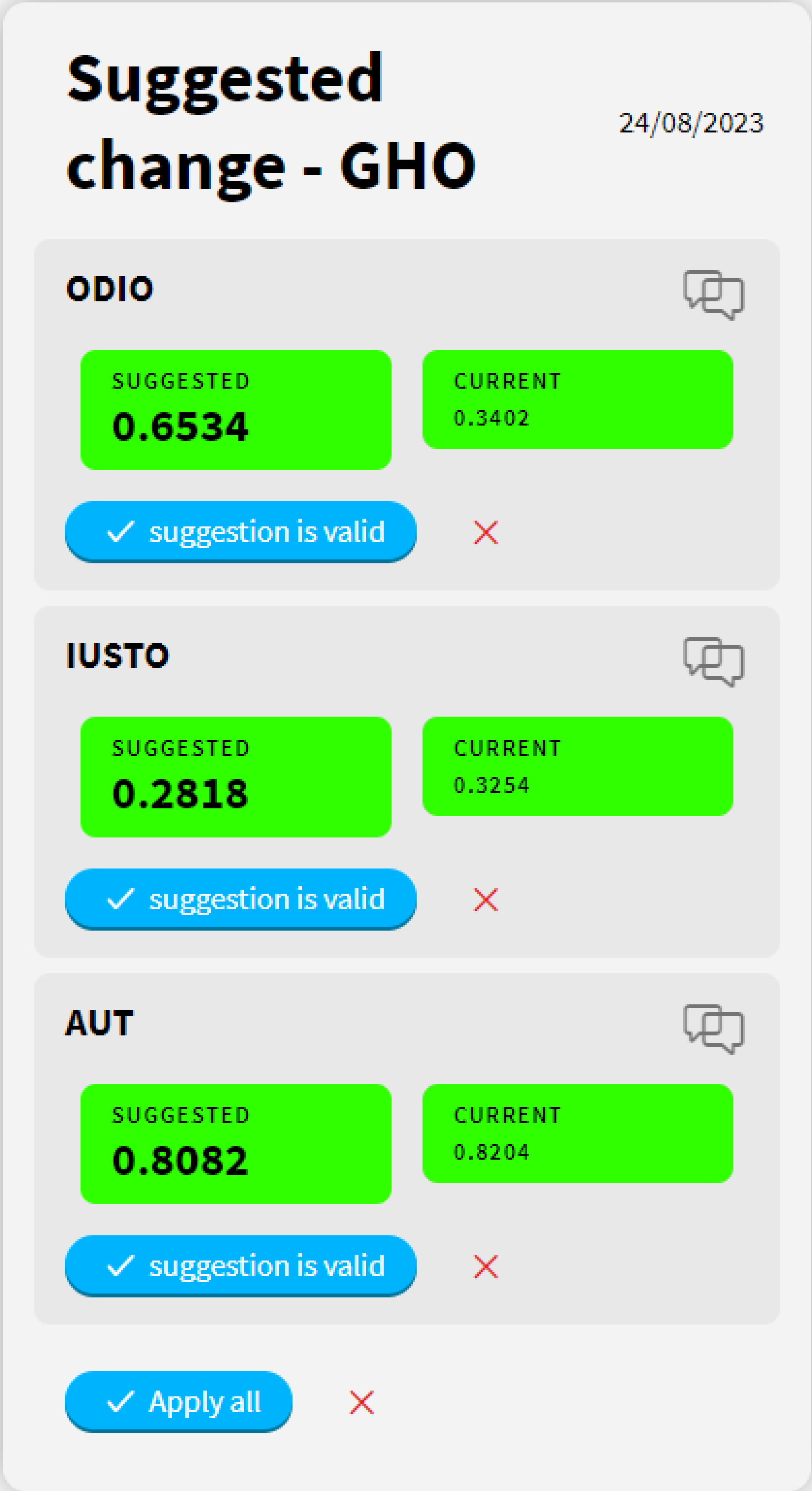

Figure 19: Conti10 widget GHO

3.1.3 CONTI3 deployment

Conti3's widget notifies the operator of what values to look at to influence the current product tension. This is to bring the product to the correct measurements. Figure 20 is an example version of the widget deployed on the continental dashboard.

Figure 20: Conti3 widget

3.1.4 CONTI7 deployment

CONTI7 widget is not being deployed at the moment of this deliverable writing due to some delays caused by the disruption of the supply chain of the critical components. This disruption is a direct result of the COVID-19 restrictions that were posed worldwide, given the systematic and modular nature of the dashboard concept used in AI-PROFICIENT. As soon as the hardware components needed for the AI service to stream the data to the CONTI7 dashboard are installed, the HMI functionality can be setup within a confined time frame, with predictable efforts and implementation flow. Some more information about the impediments within the CONTI7 use case is provided in the adjacent deliverable D4.2.

3.1.5 INEOS deployment

The INEOS mobile application is deployed on the INEOS factory floor on their mobile device. The purpose of the application is to streamline the check-in of an additive to a feeder and to allow real-time communication to the DCS. The application communicates with the TenForce backend deployed in the cloud through the internet. The deployed HMI is displayed in Figure 21. The figure shows three different screens within the flow of the application. The first screen (top of the figure) displays the HMI where the operator determines the feeder where he is at. The screen in the middle of the figure is where the label of the additive which the operator wants to add to the feeder. Lastly , the screen on the bottom of the figure is the confirm screen, where the operator has the control to change or confirm his choice and where he will receive the communication from the DCS.

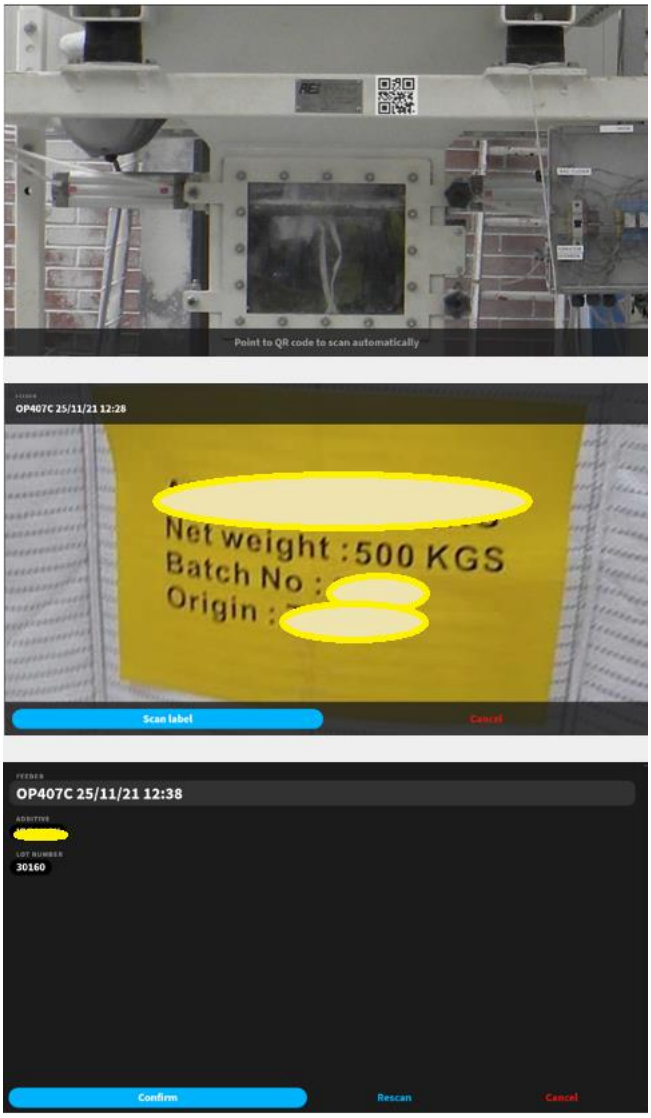

Figure 21: INEOS application screens

4 Stakeholders' access to HMI and its components

This section lists the stakeholders and their respective access to the HMI and its components for the Continental dashboard and the INEOS application. The team identified four types of stakeholders: data scientists, developers, factory managers and operators, as well as IT specialists involved from both the factory and technical partners sides.

Continental dashboard:

Data scientist	• Connects to the server via the VPN. • Has access to the backend and data
Developer	• Connects to the server via the VPN. • Has access to the HMI and the backend
Operator/manager	• Connects to the server via the VPN or through the local network. • Has access to the HMI
IT specialist	• Connects to the server via the abovementioned VPN or through the local network. • Has access to the HMI, backend and data

INEOS application:

Data scientist	• Has access to TenForce backend with his credentials. • No access to the local network with the HMI. • No access to the DCS network.
Developer	• Has access to TenForce backend with his credentials. • No access to the local network with the HMI. • No access to the DCS network.
Operator/manager	• No access to TenForce backend with his credentials. • Has access to the local network with the HMI. • No direct access to the DCS network.
IT specialist	• Has access to TenForce backend with his credentials. • Has access to the local network with the HMI. • Has access to the DCS network.

5 Conclusion

The HMIs described in this deliverable have traversed a range of technological solutions within various domains. From mobile applications and web interfaces to augmented reality machine learning, human feedback and explainable AI. Each facet still allows the user to still be in control of the user-centric design.

These solutions have provided an improved flow and allow the user to view the information from a different perspective. The machine learning services allow the user to make correlations between certain machine parameters and have a better understanding of the process interactions.

The mobile application provided a dynamic interface that employs image-driven text processing, user interaction, and communication with distributed control systems. This intricate network illustrates the fusion of diverse technologies, including image recognition, OCR, secure communication, and error handling, to create a cohesive user experience that bridges the gap between mobile applications and industrial control systems.

These HMI's collectively highlight the synergy between technology and user needs. Whether through hands-free AR interaction, web-based machinery monitoring, or image-driven mobile applications, the overarching theme is clear: technology must be harnessed to empower users, streamline processes, and optimize decision-making.

6 Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957391.