H2020 - INDUSTRIAL LEADERSHIP - Information and Communication Technologies (ICT)
ICT-38-2020: Artificial intelligence for manufacturing
Grant Agreement: 957362

Explainable Manufacturing Artificial Intelligence

XMANAI
MAKING AI UNDERSTANDABLE

WP1: Explainable AI Foundations Elaboration and XMANAI Concept Fusion

D1.2: XMANAI Concept Detailing, Initial Requirements, Usage Scenarios and Draft MVP

Deliverable Leader: Suite5

Due Date: 31/07/2021

Dissemination Level: Public

Version: F1.0

Short Abstract

This deliverable aims at bringing together the XMANAI concept by: (a) brainstorming on different user journeys in Explainable AI for business users, data scientists and data engineers, (b) eliciting the backlog of technical requirements and aligning them with the business requirements and the user journeys, (c) obtaining some early perspectives on the available manufacturing data (from the XMANAI demonstrators and open data sources), and (d) consolidating the Minimum Viable Product (MVP) that summarizes the expected features on which XMANAI shall focus (by the end of the project) for maximizing the expected added value to manufacturers while ensuring innovation from a scientific and technical perspective.

Document Log

Contributors	**Suite5**, TXT, Fraunhofer, TYRIS, POLIMI, AIDEAS, INNOVALIA, ATHENA, KBIZ, WHR, FORD, UBITECH, DBL, UNIMETRIK, CNH
Internal Reviewer 1	TYRIS
Internal Reviewer 2	WHR
Type	Report
Delivery Date	31/10/2021 (M12)

History

Versions	Description
D0.1	Initial Table of Contents
D0.2	Initial draft of Section 2 consolidating inputs during the brainstorming exercise by all partners
D0.3	Initial draft of Section 3.2 containing inputs by technical partners
D0.4	Revision of Section 3.3 consolidating and harmonizing the technical requirements
D0.5	Initial version of Section 5.1 and 5.2 for assessment by all partners
D0.6	Consolidated Section 4 with information by the Trial Handbook and contributions by all partners on Section 4.3
D0.7	Full Deliverable Draft sent for internal review
R0.1	Revision of internal reviewer 1 (TYRIS)
R0.2	Revision of internal reviewer 2 (WHR) based on R0.1
D0.8	Updated version addressing comments received during the internal review
F1.0	Final version submitted to the EC

Executive Summary

The XMANAI Deliverable D1.2 “XMANAI Concept Detailing, Initial Requirements, Usage Scenarios and Draft MVP” documents the results achieved by tasks T1.3 “Platform Requirements Elicitation, Data Acquisition and AI Scenarios” and T1.4 “XMANAI Concept Elaboration, MVP Definition and Validation”, as well as the performed work in this context. The purpose of this deliverable is to define the basic ingredients of the XMANAI concept in the form of user journeys, data and technical requirements and the preliminary features of the Minimum Viable Product (MVP) that will drive the next implementation steps of the project.

Following the patterns of the agile development methodology, the XMANAI high-level usage scenarios, noted as User Journeys, have been elaborated considering three different, but equally important, roles: the Business User, the Data Scientist and the Data Engineer. These scenarios define the AS-IS and TO-BE situation for each role in order to highlight the current challenges and the expected changes that shall be introduced through XMANAI. For each TO-BE User Journey, different phases (in total 8, across the different stakeholders) have been described along with each stakeholder’s expectations, benefits and challenges.

In terms of capturing the actual users needs, such user journeys are complemented by the collection of 134 technical requirements that have been brainstormed by all partners in the form of user stories. The mapping between the technical requirements and: (a) the different steps envisaged across the phases of the users journeys, and (b) the 46 business requirements directly gathered by the project’s demonstrators (as elaborated in the XMANAI Deliverable D6.1), has been explicitly provided. Each demonstrator has also provided the initial information related to its own data sources and data acquisition methods that will interact with the XMANAI platform and feed the AI models while a desk-based research has been performed to collect 22 complementary, open manufacturing datasets.

Finally, the process and the initial outcomes of the XMANAI Minimum Viable Product (MVP) have been defined. Based on the technical requirements, the XMANAI platform features are extracted and assessed (qualitatively and quantitatively) from the XMANAI demonstrators and the technical partners in order to be evaluated and prioritized with regard to their added value and innovation in manufacturing (from the demonstrators perspective), and their complexity, feasibility and innovation (from the technical perspective). This preliminary MVP consolidation has led to 7 “must-have” features (or epics), 16 “should-have” features, 15 “could-have” features and 7 “won't-have right now” features (that essentially inherit their prioritization to their associated technical requirements).

The results of this deliverable shall be leveraged as input to the XMANAI architecture design in “T5.1-Platform Architecture, Bundles Communication Design and APIs Definition”, and will guide the design and early development tasks of WP2 “Industrial Asset Management and Secure Asset Sharing Bundles”, WP3 “Core Artificial Intelligence Bundles for Algorithm Lifecycle Management” and WP4 “Novel Artificial Intelligence Algorithms for Industrial Data Insights Generation”. Furthermore, the activities of the tasks related to this deliverable will continue to reflect on the project’s advancements, and two more iterations of this document will follow, updating and improving the present, initial findings on M18 and M30 and shall be documented in D1.3 and D1.4, respectively.

Table of Contents

List of Figures

List of Tables

1 Introduction

The main aim of this section is to provide a brief overview of the deliverable.

1.1 XMANAI Project Overview

Despite the indisputable benefits that Artificial Intelligence (AI) can bring in society and in any industrial activity, humans typically have little insight about AI itself and even less concerning the knowledge on how AI systems make any decisions or predictions due to the so-called “black-box effect”. Many of the machine learning/deep learning algorithms are opaque and not possible to be examined after their execution to understand how and why a decision has been made. In this context, to increase trust in AI systems, XMANAI aims at rendering humans (especially business experts from the manufacturing domain) capable of fully understanding how decisions have been reached and what has influenced them.

Building on the latest AI advancements and technological breakthroughs, XMANAI shall focus its research activities on Explainable AI (XAI) in order to make the AI models, step-by-step understandable and actionable at multiple layers (data-model-results). The project will deliver "glass box" AI models that are explainable to a "human-in-the-loop", without greatly sacrificing AI performance. With appropriate methods and techniques to overcome data scientists’ pains such as lifecycle management, security and trusted sharing of complex AI assets (including data and AI models), XMANAI provides the tools to navigate the AI’s “transparency paradox” and therefore:

(a) accelerates business adoption addressing the problematic that “if manufacturers do not understand why/how a decision/prediction is reached, they will not adopt or enforce it”, and
(b) fosters improved human/machine intelligence collaboration in manufacturing decision making, while ensuring regulatory compliance.

XMANAI aims to design, develop and deploy a **novel Explainable AI Platform** powered by explainable AI models that inspire trust, augment human cognition and solve concrete manufacturing problems with value-based explanations. Adopting the mentality that “AI systems should think like humans, act like humans, think rationally, and act rationally”, a catalogue of **hybrid and graph AI models** is built, fine-tuned and validated in XMANAI at 2 levels: (i) baseline AI models that will be reusable to address any manufacturing problem, and (ii) trained AI models that have been fine-tuned for the different problems that the XMANAI demonstrators’ target. A bundle of **innovative manufacturing applications and services** are also built on top of the XMANAI Explainable AI Platform, leveraging the XMANAI catalogue of baseline and trained AI models.

XMANAI will validate its AI platform, its catalogue of hybrid and graph AI models and its manufacturing apps in **4 realistic, exemplary manufacturing demonstrators** with high impact in: (a) optimizing performance and manufacturing products’ and processes’ quality, (b) accurately forecasting product demand, (c) production optimization and predictive maintenance, and (d) enabling agile planning processes. Through a scalable approach towards Explainable and Trustful AI as dictated and supported in XMANAI, manufacturers will be able to develop a robust AI capability that is less artificial and more intelligent at human and corporate levels in a win-win manner.

1.2 Deliverable Purpose and Scope

This deliverable documents the results of tasks T1.3 “Platform Requirements Elicitation, Data Acquisition and AI Scenarios” and T1.4 “XMANAI Concept Elaboration, MVP Definition and Validation” for the first iteration of WP1 “Explainable AI Foundations Elaboration and XMANAI Concept Fusion”.

Its main purpose is to elaborate in detail on the XMANAI concept as a novel approach for using explainable AI in manufacturing operations, as well as to provide an initial, high-level, version of the XMANAI Minimum Viable Product (MVP) to be used as a guide for the research and development activities of the project. To achieve these goals, the document has first to provide the key ingredients:

a) The User Journeys for all involved roles, presenting the AS-IS and TO-BE processes along with the expected benefits and challenges,
b) The initial Technical Requirements regarding the XMANAI Platform, in the form of User Stories, and
c) An early profiling of the data sources to be utilised in XMANAI, along with their requirements.

The functionalities derived from this process are carefully refined and prioritized, so that the most valuable and feasible features form the initial XMANAI MVP. This MVP will be continuously updated during the course of the project, in order to include new feature requests or research findings and follow the project's advancements. For this reason, two more iterations of this document will follow with D1.3 and D1.4, that shall update and improve on the initial findings on M18 and M30, respectively.

1.3 Impact and Target Audiences

The results presented in this document target mainly the technical users that develop the XMANAI Platform, as well as the researchers who support the solution. The business users are also influenced by the content, since in essence it describes, albeit in draft format, the XMANAI product to be built.

The resulted MVP is meant to act as the driving force behind the technical discussions that will follow and become the core technical blueprint on top of which the design of the architecture and the implementation of the platform will be based. Therefore, the impact of this deliverable, as well as of the iterations to follow, can be regarded as very significant for the project on a technical level.

1.4 Deliverable Methodology

The information reported in this deliverable has been produced by the consortium members following the methodology described below:

I. **User Journeys Elaboration**: User Journeys have been derived from repeated brainstorming sessions with the stakeholders using Miro boards[1]. All partners provided their ideas in sticky notes regarding the actions, expectations, benefits, and challenges per XMANAI phase for each role: Business User, Data Engineer, Data Scientist. The information was then collected, organised and aggregated into refined workflows for each of the high-level user roles (Business User, Data Engineer, Data Scientist).
II. **Technical Requirements Elicitation**: The technical requirements have been derived, on the same fashion through brainstorming using the Miro boards, with repeated iterations with the technical partners of the consortium. Each requirement was connected to the related task(s) of the project and the respective Task leader was responsible for refining the available suggestions and filling-in any gaps. A draft list of user stories was then consolidated and compared to the business requirements coming from "T6.1- Demonstrators Requirements Elicitation" to make sure the business needs have been considered in full.
III. **Data Requirements Extraction**: The data requirements have been derived using online questionnaires (i.e. Trial Handbook Chapter 3) and interviews with demonstrator partners where each one described the available data sources, their features and limitations, as well as the means of acquisition for the respective demonstration scenario.
IV. **MVP Definition**: The wide range of technical requirements have been grouped and consolidated into different MVP features, that have been assessed for: (a) their added value (by demonstrator partners), and (b) their feasibility (by technical partners) in order to produce a prioritized list of features and functionalities as part of the draft XMANAI MVP. The draft MVP does not provide the prioritization only for the derived MVP features but also indirectly for the whole technical requirements backlog.

[1] https://miro.com

1.5 Dependencies in XMANAI and Supporting Documents

As already stated, the XMANAI Deliverable D1.2 reports the results of Tasks T1.3 "Platform Requirements Elicitation, Data Acquisition and AI Scenarios" and T1.4 "XMANAI Concept Elaboration, MVP Definition and Validation" of WP1 "Explainable AI Foundations Elaboration and XMANAI Concept Fusion". Both of these tasks make use of the outcome of the tasks T1.1 "Explainable AI and Graph Machine Learning Analytics State-of-Play" and T1.2 "Human Aspects in Decision Making and AI" (reported in D1.1). Task 1.3, in particular, also compares the technical requirements gathered from technical partners, with the business requirements collected in the framework of T6.1 "Demonstrators Requirements Elicitation". Figure 1-1 depicts these relationships, as well as the tasks to be supported by the results contained in this deliverable.

More specifically, this deliverable is expected to primary support the activities of the architectural design task in "T5.1- Platform Architecture, Bundles Communication Design and APIs Definition", but will also be of assistance for the implementation of all "WP2 – Industrial Asset Management and Secure Asset Sharing Bundles" and "WP3 - Core Artificial Intelligence Bundles for Algorithm Lifecycle Management" activities. It needs to be noted that the data requirements exercise also provides valuable inputs to the "WP4 - Novel Artificial Intelligence Algorithms for Industrial Data Insights Generation" activities.

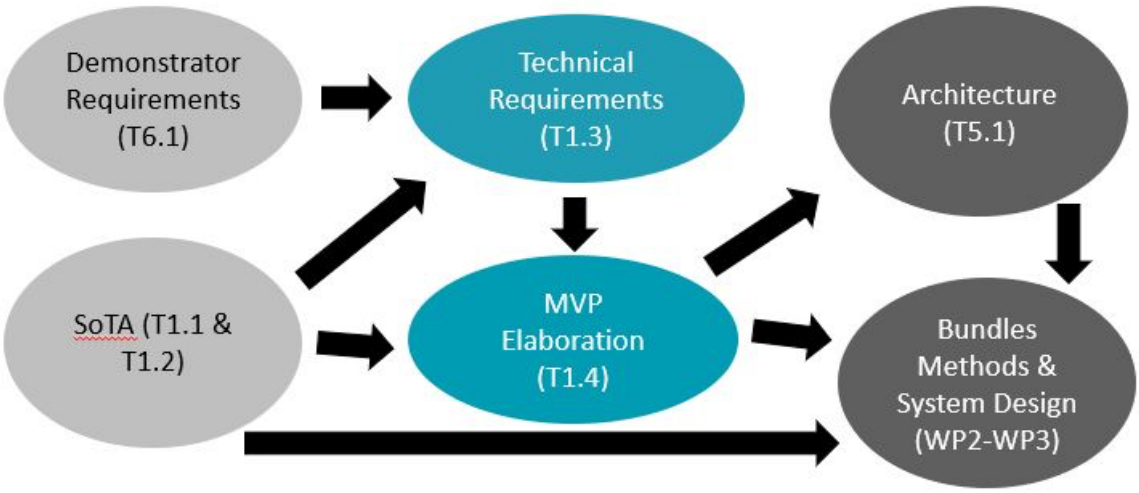


Figure 1-1: T1.3 & T1.4 Task Dependencies in XMANAI

1.6 Document Structure

The contents of this document are structured in sections as follows:

- Section 2 presents the XMANAI workflows through the concept of User Journeys and is divided in phases according to the user Role (Business User, Data Engineer, Data Scientist), with each phase having its own specific steps and features.
- Section 3 focuses on the technical requirements as expressed by business experts and data science professionals of the consortium partners in the form of User Stories.
- Section 4 presents possible data sources to be used for the XMANAI AI models and identifies relevant data-related requirements.
- Section 5 presents the preliminary consolidation of the XMANAI Minimum Viable Product (MVP) and provides an initial collection and internal assessment of the platform features.
- Section 6 concludes this deliverable and provides future steps.

2 Explainable AI User Journeys

This section introduces the current workflows the targeted stakeholders / roles of XMANAI follow and the user experience they are expected to have in XMANAI.

2.1 Stakeholders

The different stakeholders that are expected to utilize the XMANAI project results focusing on Explainable AI for the manufacturing industry, are classified into 3 high-level categories:

- **Business Users**: Domain experts within a manufacturer that specialize into the operations of different departments (e.g. production, marketing, sales, IT, etc.) and who need to understand the results of an analysis in a tangible - for them - manner. Business Users are also responsible for assessing the validity of the results and explainations from a business persepective. In XMANAI, the targeted business users typically vary per demonstrator, as described in detail in Deliverable D6.1.
- **Data Scientists**: Data experts that use scientific methods, processes and algorithms to explore and extract knowledge from data, understand the problem at hand, create AI models and derive actionable insights from data in different application domains. Data Scientists are also responsible for ensuring - from a scientific perspective - the trustability and robustness of the results, both the predictions per se and their associated explanations.
- **Data Engineers**: Software engineering experts with a strong data background that are responsible for building the necessary underlying infrastructure to collect and prepare data, and to deploy AI models and solutions to analyze data in a scalable manner. Data engineers also ensure the integrity of the data and all their associated security aspects.

2.2 Business User Journey

The Explainable AI Business User Journey describes how domain experts from the manufacturing industry currently operate in their everyday work (as-is-situation) and what is the expected to-be situation with the XMANAI Platform.

2.2.1 AS-IS Situation

While the majority of the manufacturers today have some form of familiarity with big data - how to collect it, how to store and access data, how to get some insights from an analysis, their focus has been mainly on manufacturing technology and its physical manifestation with little emphasis on a holistic digital transformation and AI strategy on how to create added value out of data, analysing them or designing decision making strategies around them (Gerdeman, 2017; Lenz et al, 2018). Still, the factory production or market sales heavily rely on human experience and empirical knowledge.

For example, regarding data ingestion in the XMANAI Demonstrators, Ford still works with manual mechanisms that slow down the analysis and decision-making processes. This leads to numerous bottlenecks and weak points that make it impossible to react quickly to various production anomalies, which affects the entire plant's performance. In CNH, the unplanned stoppage and quality data are taken in a manual way, without automation or digital extrapolation, and maintenance operations are managed without connection to digital tools, based on empirical decision making.

Even in cases where things are getting automated and data warehouses and AI tools are in place, the power of data analytics is not yet fully harnessed. The Whirlpool's demand forecasting process, for instance, suffers from a set of challenges related to the low prediction accuracy and limited insights on the predictions. Another challenge is the very high complexity embedded into the demand management process, since many factors and variables have multiple correlations that are not always clear. This poses some limitations to humans trying to understand, control and even optimize this process. The lack of a collaboration tool that could allow business users to interact and give feedback to data scientists makes the demand management process even more cumbersome.

Re-using information and processed datasets is also lacking in manufacturing. In Unimetrik's metrological work, the measurement strategy of a part is defined from scratch for each job. In the case of reusing values from another measurement, the data to program the machine is entered by hand, which leads to a considerable loss of time.

In general, the **AS-IS** workflow for business users in manufacturing follows a more or less valid approach: the company collects all data available within their organisation, either from sensors, operational log files or manual feed, and use some modern commercial BI tool to extract insights or make forecasts.

The main challenges that are currently observed can be summarized as:

- Obtaining and maintaining high quality data in an effortless manner;
- Efficiently enriching data by joining data sources from different departments (like sales with marketing promotions) or from other organisations or open sources;
- Sharing a common understanding for the meaning and the processes that the data convey across different roles;
- Contributing to the iterative training, fine-tuning and evaluation of AI models in a collaborative way with the data scientists;
- Lack of any form of explainability regarding the various AI results, a fact that makes business users struggling to put their trust (and investment) on any automated predictions/decisions.

2.2.2 To-BE Situation with XMANAI

The Business User journey consists of three different phases: Phase 1 is related to how data can be provided and communicated in an efficient way, from the business user perspective, towards AI Preparation. Phase 2 deals with the iterative understanding and evaluation of AI models and analytics results for AI experimentation. The last phase, Phase 3, refers to the insights gained by the AI results and how these can increase business efficiency while AI is in "production".

As all XAI User Journeys, the Business User journey was created collaboratively by the XMANAI partners as displayed through the initial and consolidated Miro boards in Figure 2-1.

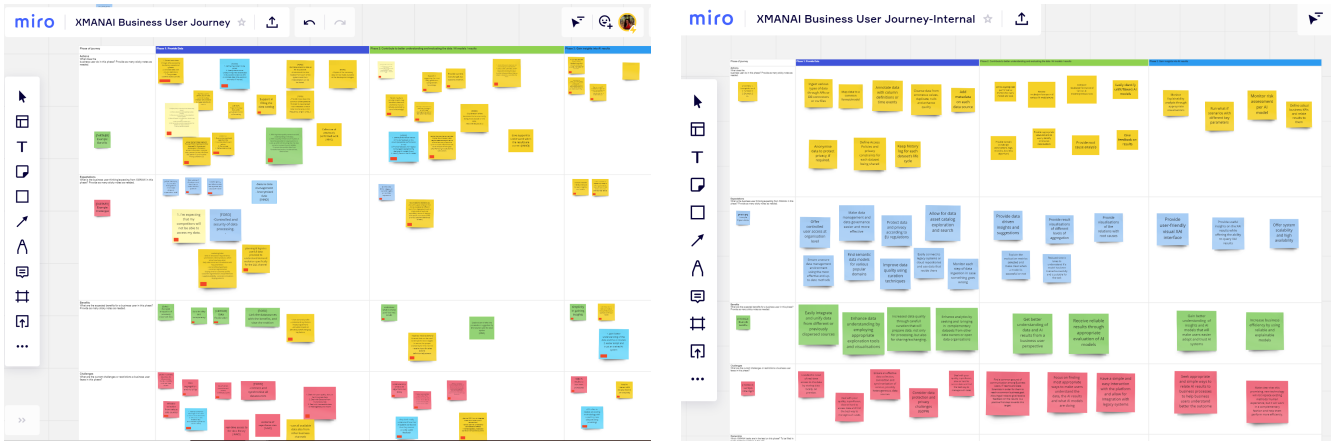


Figure 2-1: XAI Business User Journey (left: initial Miro board; right: consolidated Miro board)

Phase 1: AI Preparation - Provide and Understand Data

The AI journey for the Business User (with a strong understanding of the data within the manufacturer) commences with the provision and collection of data. Data ingestion from various sources, cleansing erroneous or duplicate values, and mapping to a common data model to provide insights into the data are the main AI preparatory steps encountered during this phase.

During this phase, the expectations of any business user from the XMANAI platform can be summarized into:

- Data protection in terms of a secure data management environment where data are easily ingested, while privacy is protected.
- Data interoperability, considering both its semantic and syntactic perspectives, where data are expected to be able to be exchanged and integrated in an efficient manner.
- Data quality that can be increased through proper data curation, but also through the effective handling of missing information.
- Data sharing to ensure that all involved stakeholders can have access to the data and all its accompanying information in an IPR-respectful manner and considering the access policies that the business users have defined.

The benefits from using an XAI-oriented approach in the XMANAI platform, that fulfils the aforementioned expectations, are the easy and secure integration of data from different or previously dispersed sources, the enhanced and quicker understanding of the available information, and the increased data quality that shall eventually lead to improved insights into the actual business operations.

From a business user perspective, the main challenges that need to be addressed in this phase concern: (a) effective handling of connection, synchronisation and interoperability issues across the various data sources; (b) dealing with protection and privacy regulations (such as GDPR), but also with the varying security requirements and processes that are internally applied per manufacturer; (c) managing low quality or superfluous datasets in the most efficient manner.

Table 2-1: Business User Journey Phase 1 - AI Preparation

Phase 1: AI Preparation - Provide and Understand Data	
Actions	Ingest data in different ways (e.g. through APIs or DB connectors or as csv files)
	Map data to a common data model
	Annotate data with column definitions or data types
	Cleanse data from erroneous values, duplicate, nulls and enhance quality
	Add metadata on each data source
	Anonymise data to protect privacy, if required
	Define access policies and privacy constraints for each dataset
	Keep history log for each dataset's life cycle
Expectations	Offer controlled user access at organisation level
	Make data management and data governance (across previously dispersed data sources) easier and more effective
	Protect data and privacy according to EU regulations
	Allow for data asset catalogue exploration and search
	Ensure a secure data management environment using the most effective and up-to-date methods
	Enable data providers to describe their data using relevant semantic data models
	Improve data quality using data curation techniques

	Easily connect to legacy systems or cloud repositories through the APIs that they expose and use data that reside there
	Monitor each step of data ingestion in case something goes wrong
Benefits	Easily integrate and unify data from different or previously dispersed sources
	Enhance data understanding by employing appropriate exploration tools and visualisations
	Increased data quality through careful curation that will prepare data, not only for processing, but also for sharing/exchanging
	Enhance analytics by seeking and bringing in complementary datasets from other data owners or open data organisations
Challenges	Handle access to confidential data by storing data on premise (i.e. on private cloud infrastructures, on private servers).
	Deal with poor quality, superfluous, slow or hard to access data and find the best way to manage such cases
	Ensure an effective data collection, connection and synchronisation of various, possibly heterogeneous, data sources
	Consider data protection and privacy challenges (GDPR)
	Deal with low quality or missing metadata. Enable user to improve the quality of metadata manually, and try to support the user with some automatic metadata generation from data

Phase 2: AI Experimentation - Contribute to better understanding and evaluating the AI models / pipelines / results

The second phase of the Business User journey refers to the process of providing business/domain knowledge in order to iteratively improve the understanding of the problem at hand, the creation of appropriate analytics models and the evaluation of ML/DL models and their results.

In this phase, the business user expects to be able to inspect the AI models, understand their reasoning and provide feedback on the analytics results along with their explanations. For this to happen, proper and easy to understand visualisations should be provided, considering the business perspective, as well as coherent explanations on the selected evaluation metrics and on the model performance.

The most significant benefit of this stage is for the business users to gain insights into the analytics models results and explanations, as well as to tell the difference between a successfully trained AI model and a failed one. Getting reliable and comprehensive results during the training and testing phase will increase their trust towards AI, which is a core challenge for any business user. Collaboration and finding of a common ground among business users, IT teams and data scientists will help confront this challenge.

Table 2-2: Business User Journey Phase 2 - AI Experimentation

Phase 2: AI Experimentation - Contribute to better understanding and evaluating the data / AI models / results	
Actions	Define appropriate performance metrics for each model and task
	Review results/performance/explanations of various AI models/runs
	Compare results/performance/explanations of various AI models/previous runs
	Easily identify unfit/flawed AI models

	Provide appropriate visualisations to easily identify erroneous data/outliers, as well as results
	Provide domain knowledge (descriptions, tags, relations, business objectives) for better explaining results
	Perform root cause analysis
	Ask exploratory questions over the training phase or the testing results
	Give targeted feedback on results and explanations (based on domain knowledge)
Expectations	Gain data-driven insights and suggestions for business problems
	Understand the analytics results (predictions of a ML/DL model or pipeline)
	Configure appropriate result visualisations at different levels of aggregation
	Provide annotations for the relations of results with root causes
	Understand the evaluation metrics selected by a data scientist, whether a model has been trained successfully and is suitable for the task
Benefits	Get better understanding of AI results from a business user perspective
	Receive reliable and actionable results along with appropriate explanations, upon an appropriate evaluation of AI models
Challenges	Find a common ground of communication among Business Users, IT teams and Data Scientists in order for them to reach a common business goal
	Allowing all roles to give/receive feedback on the AI models/pipelines in a fruitful and productive manner
	Focus on finding the most appropriate ways to make business users understand the AI results and what AI models are doing
	Have a simple and easy interaction with the platform while allowing for integration with legacy systems

Phase 3: AI Insights - Understand AI results

Gaining appropriate insights into AI results while an AI system is in production is the third and last phase of this journey. A business user should be provided with proper tools that enhance the explainability of the results, monitor risk assessment and relate the outcomes with real business KPIs. Leveraging different explainability techniques (e.g. explanations by example or running what-if scenarios with different key parameters and appropriate visualisations) will certainly help in this direction.

From this phase, the business user expects to get useful insights, but also to be allowed to query and interact with the AI predictions in a user-friendly, business oriented, interface. The high availability and scalability of the platform are also two factors that belong in the expectations pool for the business user.

Extracting tangible and actionable knowledge from AI results, along with their explanation, will be beneficial for any manufacturer that aims to increase business efficiency through data processing and analytics. Moreover, it will eventually increase adoption of AI in their work.

Nevertheless, it should be made clear that AI will not directly replace existing methods or human experience but will act in a complementary way, building on and augmenting the human experience, that will help all actors involved to perform more efficiently and in a trusted manner.

Table 2-3: Business User Journey Phase 3 - AI Insights

Phase 3: AI Insights - Understand AI results	
Actions	Receive explainable results in an appropriate way (e.g. through visualisations, notifications)
	Run what-if scenarios with different key parameters
	Monitor risk assessment per AI model
	Define actual business KPIs to which AI results can be related
Expectations	Utilize user-friendly and intuitive Explainable AI interfaces depending on the business problem at hand (dashboards, mobile apps)
	Provide useful insights on the AI results in a user-understandable manner
	Interact with the AI results by querying or requesting further explanations
	Offer system scalability and high availability
Benefits	Gain better understanding of AI decisions that will make users easier adopt and trust AI systems
	Increase business efficiency by using reliable and explainable models
Challenges	Seek appropriate and simple ways to relate AI results to business processes to help business users understand better the outcome
	Make clear that this promising AI technology will not replace existing methods/ human experience, but it will work in a complementary fashion and help the business users perform more efficiently

2.3 Data Scientist Journey

The Explainable AI Data Scientist Journey describes how data scientists, working within a manufacturer or externally, currently operate in their everyday work (as-is-situation) and what is the expected to-be situation with the XMANAI Platform.

2.3.1 AS-IS Situation

Plant level engineering is one area where data scientists are rarely found (Posey, 2019), which is quite unexpected considering the vast amount of data collected by sensors at the shop floor. Nevertheless, in the last few years more and more manufacturing enterprises have started to hire data scientists or external IT companies capable of building data pipelines and analysing data at scale.

Typically, the goal of a data scientist is to derive useful insights from data that may answer questions of interest for solving a specific business problem. To achieve this, the data scientist follows an often non-linear and extremely iterative workflow that consists of the following activities (see Figure 2-2): First, the data scientist needs to get the definition of the problem which needs to be thoroughly understood. The second phase is to collect/acquire the relevant data, which, in the next phase need to be explored and manipulated in order to view potential correlations, patterns and to develop a hypothesis (and the appropriate dataset) that could be tested in the next phases. Once a hypothesis is generated, the data scientist typically builds an AI model that learns from data (through a training process) and is then able to generalize against unseen examples of the same problem, which needs to be validated and evaluated. The last stage involves the study and presentation of the results which in many cases requires special communication skills, especially when the audience is not very experienced in Data Science.

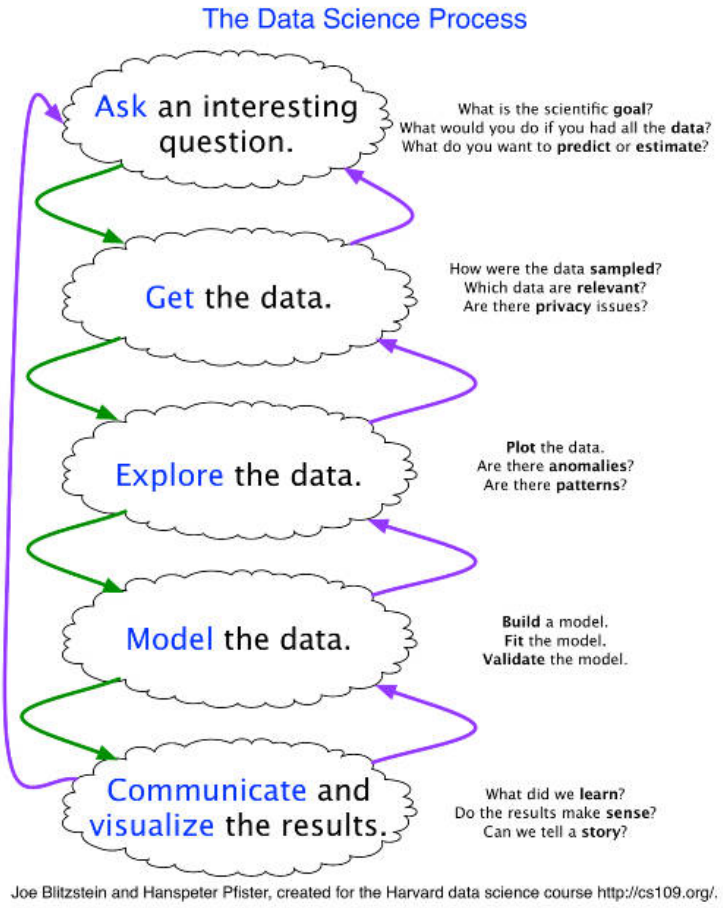


Figure 2-2: The Data Science Process as per Blitzstein and Pfister

This, as well as several other approaches that can be found in the literature up until these days, were developed using the Cross-Industry Standard Process for Data Mining (CRISP-DM) framework as a basis (Martínez-Plumed et al, 2019). One such case is the CASP-DM model, that addresses many challenges of modern machine learning for context change and model reuse handling (Martínez-Plumed et al, 2017) as depicted in Figure 2-2.

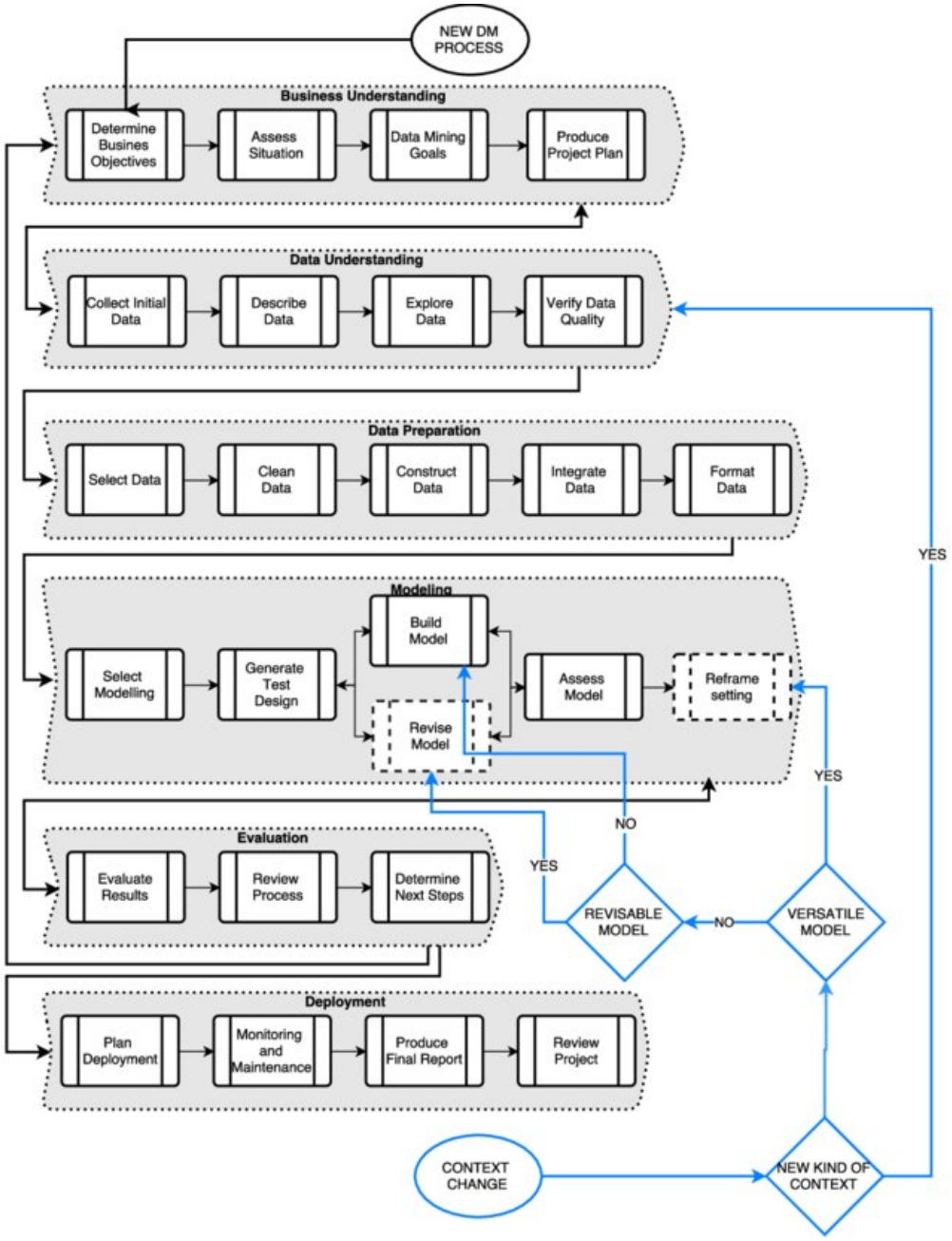


Figure 2-3: Complete view of CASP-DM model and tasks (Martínez-Plumed et al, 2017)

Another interesting case, as well as the most recent, is the framework proposed by Schmidt and Sun in 2018. The so-called Agile-facilitated KD model introduces agile practices for conducting activities in each sub-process, including: Business understanding, Data selection, Preprocessing, Transformation, Interpretation/ Evaluation, Deployment.

Both the CASP-DM model and the Agile-facilitated KD model support a deployment phase, as part of a complete data science process, even though it is noted by the authors that deploying an AI model or pipeline is the most challenging step for several factors (i.e. personnel, funding, support to senior managers, etc.).

It is noteworthy to notice that even the more recent of all these approaches, lack the AI explainability part, which nowadays has become an increasingly essential part for a modern data scientist workflow.

2.3.2 To-BE Situation with XMANAI

The user journey of a data scientist consists of five different phases. Phase 1 refers to understanding the available data and the problem that needs to be handled as an AI preparatory phase. Phase 2 covers the data preparation process including the proper manipulation of the problematic data cases. Phase 3 refers to the design of the pipeline, in which phase the data scientists need to be able to collaborate and share their ideas and configurations with other users as well and appropriately train the models included in the pipeline. Phases 2 and 3 can be characterized as the core of AI experimentation. In terms of AI Insights, Phase 4 involves all the actions that are required so that the results and the models themselves are explained, in order to allow different users to comprehend how the outputs occurred. Finally, the last phase refers to the evaluation methods for assessing the performance of the models, so that the changes that will improve the performance can be recognized and made.

As all XAI User Journeys, the Data Scientist journey was created collaboratively by the XMANAI partners as displayed through the consolidated Miro boards in Figure 2-4.

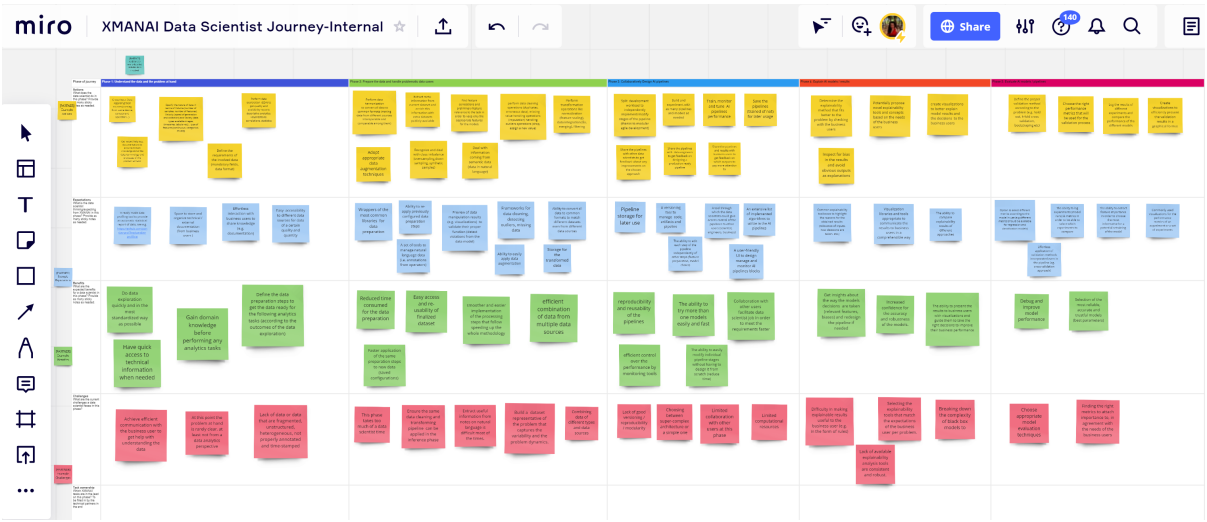


Figure 2-4: XAI Data Scientist Journey (consolidated Miro board)

Phase 1: AI Preparation - Understand the data and the problem at hand

During phase 1, data scientists will need access to tools and technologies that will help them get access to the data, get familiar with the data and understand in depth all the aspects of the problem at hand. To that end, the XMANAI platform is expected to provide all the necessary tools for the data scientists. Data profiling is essential in order to obtain an understanding over the data through statistical reports of the data and their nature, such as the number of samples, the features and the types of data (images, timeseries, tabular). Data exploration is expected to contribute to exploratory data analysis through visualizations and correlations among different features in the data. In addition, the platform should facilitate the access to different data sources (provided by data owners) and ensure that technical knowledge coming from the business users can be effectively communicated to the data scientists. In this way, data scientists will be able to access necessary information to comprehend the data and the problem.

Using the XMANAI platform, the data scientists will be benefitted in many ways. They will be able to quickly explore the data and acquire quality and availability reports in a standardized way. They will have access to technical information at will, and will gain domain knowledge before proceeding to the analytic tasks. After completing phase 1, data scientists will have a clear view of the data preparation steps, that are required to get the data ready for the future tasks.

However, typically the business problem at hand is not well defined from a data analytics perspective and thus it is not clear for data scientists. In addition to that, efficient communication with business users, in order to share domain knowledge and explanations on what the data represent is usually difficult to achieve. Finally, there is often lack of quality data, as they might be fragmented, unstructured, heterogeneous and not properly annotated. The XMANAI platform will try to address all these challenges and minimize their effect as much as possible for the data scientists and the initial time-consuming analysis of the data.

Table 2-4: Data Scientist Journey Phase 1 – AI Preparation

Phase 1: AI Preparation - Understand the data and the problem at hand	
Actions	Find the appropriate data and group them in various ways, e.g. regarding their sources (coming from same device, component, operation...)
	Acquire domain knowledge about the data, terminology and processes of the problem at hand
	Specify the nature of data in terms of Volume (number of samples, number of features), Velocity (speed of generation and collection) and Variety (data types, available images, timeseries, tabular etc), type of features (continuous, categorical, mixed))
	Map the involved data (mandatory fields, data format) to an appropriate data model
	Perform data exploration to get quality and availability reports, descriptive analytics, visualizations, correlations, statistics
Expectations	Automatic generation of statistical reports of data
	Effortless interaction with business users to share knowledge (e.g., documentation)
	Easy accessibility to different data sources for data of a certain quality and quantity
Benefits	Quick data exploration in the most standardized possible way
	Gain domain knowledge before performing any analytics tasks
	Define the data preparation steps to get the data ready for the following analytics tasks (according to the outcomes of the data exploration)
Challenges	Achieve efficient communication with the business user to get help with understanding the data
	At this point the problem at hand is rarely clear, at least not from a data analytics perspective
	Lack of data or data that are fragmented, unstructured, heterogeneous, not properly annotated and time-stamped

Phase 2: AI Experimentation - Prepare the data and handle problematic data cases

During phase 2, data scientists will perform all the required data preparation steps to get the data ready for the next phase of training the models. Thus, data scientists need to convert all data to common formats and match / join different datasets from different data sources. By performing this data harmonization step, data from different sources will be made interoperable and comparable. Frameworks for data cleaning, detecting outliers and missing data operations are also necessary, for handling duplicates, erroneous or missing data as well as outliers that need to be erased or assigned a new value. In addition, common libraries for data preparation, pre-processing and visualization should be at the disposal of a data scientist in order to easily perform filtering, merging, normalization and scaling as well as feature relevance recognition. Through visualizations data scientists will be able to validate the outcomes of data preparation steps. In cases where data quantity is not adequate or class imbalance is present, data scientists should be able to perform the appropriate data

augmentation techniques (modified copies of existing data or new synthetic samples) in an automatic or manual manner. Finally, a storage space is expected to exist for the transformed data, as well as for the data preparation pipeline, so that data scientists can easily re-apply the same data preparation steps to new incoming data.

Having all the necessary tools available through the XMANAI platform, data scientists will manage to reduce the overall time for the data preparation phase. With the help of the available libraries and tools, the implementation of the processing steps will be smooth and easy, avoiding the appearance of any blocking data-related issues. In addition, the efficient combination of data from different data sources will allow the concurrent data processing, in order to extract valuable conclusions about its importance and relevance to the task. Storing the transformed data will permit data scientists to directly access them and proceed to the next phase of training the models, instead of repeating the data preparation steps. Stored configurations (data preparation pipelines) are of the same importance, especially during inference, where the preparation steps applied during training should also be applied to the unseen data. Saving the data preparation pipelines, will ensure consistency and robustness of the system, as well as reduced application time.

The data preparation phase is the most challenging in the majority of analytics projects and it takes significant time by the data scientist. Combining data of different types and sources is not always straightforward. In addition, it requires a lot of effort to build a dataset that is representative of the task and captures the variability and the problem's dynamics. There are cases where the data need a lot of preprocessing before they reach an accepted form for the next steps, and it is often not easy to extract useful information from them. Moreover, ensuring that the same data cleaning and transforming pipeline will be applied in the inference phase is a laborious task that requires a lot of attention.

Table 2-5: Data Scientist Journey Phase 2 – AI Experimentation

Phase 2: AI Experimentation - Prepare the data and handle problematic data cases	
Actions	Perform data harmonization to convert all data to common formats (making data from different sources interoperable and comparable among them)
	Find feature correlations and preliminary feature relevance to the task in order to keep only the appropriate features for the models
	Perform data cleaning operations (duplicates, erroneous data), missing value handling operations (imputation), handling outliers operations (drop, assign a new value)
	Perform transformation operations like normalization (feature scaling), data integration (e.g. appending rows or appending columns), filtering
	Adopt appropriate data augmentation techniques
	Recognize and deal with class imbalance (oversampling, down-sampling, synthetic samples)
	Search for relevant datasets that are publicly available based on a dataset's metadata
Expectations	Utilize the most common libraries for data preparation (e.g. pandas) with which a data scientist is already familiar
	Re-apply previously configured data preparation steps
	Preview of data manipulation results (e.g., visualizations) to validate their proper function (detect violations from the data model)
	Different methods/options for data cleaning, detecting outliers, handling missing data
	Easily apply data augmentation
	Join/merge different datasets from different data sources

	Storage and easy access to the transformed data
Benefits	Reduced time consumed for the data preparation
	Easy access and re-usability of finalized dataset
	Smoother and easier implementation of the processing steps speeding up the whole AI process
	Efficient combination of data from multiple data sources
	Faster application of the same preparation steps to new data (saved configurations)
Challenges	Data preparation in principle takes too much of a data scientist time
	Ensure the same data cleaning and transforming pipeline can be applied in the inference phase
	Build a dataset representative of the problem that captures the variability and the problem dynamics
	Combining data of different types and data sources

Phase 3: AI Experimentation - Collaboratively Design AI pipelines

Phase 3 involves all the steps required for the design and training of the AI models. For this phase, the XMANAI platform is expected to provide a user-friendly UI to design, manage and monitor AI pipelines and an extensive catalogue of ready-to-use algorithms, including both baseline algorithms and pre-trained models. Data scientists will be able to design, experiment, train and tune the performance of as many pipelines and models as needed. Each stage of the pipelines should be implemented/modified independently of other stages, so that the users will be able to split the implementation workload and benefit from modular agile development. To ensure an efficient way to monitor the models and the training parameters, as well as to compare the results of the models, versioning functionalities are required. As in most of the phases, XMANAI should provide a pipeline storage space, from which the trained or not pipelines could be easily retrieved for editing or (further) training. Finally, since the feedback of other users is vital, it is important to ensure that the data scientists could share and grant access to the pipelines to other users (scientists, engineers, business).

The XMANAI platform will assist data scientists in different ways. Being able to easily modify individual pipeline stages without having to design it from scratch, reduces the time spend for pipeline design. Versioning shall ensure efficient control over the performance of the models and keep records that the data scientists can refer to for a more meaningful manipulation over the models' parameters. In addition, reproducibility and reusability of the pipelines is achieved both by providing a pipeline storage space and versioning. Finally, being able to collaborate with other stakeholders for the pipeline implementation, helps data scientists to meet faster their own requirements and build pipelines that meet also the requirements of the business users. More specifically, getting feedback from other data scientists can help improve the chosen approach in terms of the performance, while the insights of data engineers can help build a production-ready pipeline and the business users can indicate to which outputs should be given more attention to.

The main challenges that data scientists face at this point are mainly related to the difficulty in choosing between a complex model or a simpler one. For this reason, keeping track of the performance of each model is important, as well as being able to collaborate with other users. However, most of the times communication with others is limited at this phase. The same is true for the case of good versioning. Monitoring and managing the pipelines is often absent, which also affects their reproducibility.

Table 2-6: Data Scientist Journey Phase 3 – AI Experimentation

Phase 3: AI Experimentation - Collaboratively Design AI pipelines	
Actions	Split development workload to independently implement/modify stages of the pipeline (thanks to modular agile development)
	Build and experiment with as many pipelines and models as needed
	Train, monitor and tune AI pipelines performance
	Save the pipelines (trained of not) for later usage
	Share the pipelines with other data scientists to get feedback about any improvements on the chosen approach
	Share the pipelines with data engineers to get feedback on designing a production ready pipeline
	Share the pipelines and results with business users to get feedback on which outputs to pay more attention to
Expectations	Pipeline storage for later use
	Versioning management regarding code, artifacts and pipelines
	Provide access to pipelines to other users (scientist, engineers, business)
	Library of algorithms to utilize in the AI pipelines
	Independence in the pipeline steps allowing editing each step without affecting other steps (feature preparation, model choice)
	A user-friendly UI to design, manage and monitor AI pipelines
Benefits	Reproducibility and reusability of the pipelines
	Easily try more than one models in a pipeline
	Collaboration with other users in order to meet the requirements faster
	Efficient control over the performance of a model / pipeline
	Easily modify individual pipeline stages without having to design it from scratch (reduce time)
Challenges	Lack of good versioning / reproducibility / modularity
	Choosing between complex models or simpler ones
	Limited collaboration with other users (especially business users)
	Limited computational resources

Phase 4: AI Experimentation - Explain AI models / results

In this phase, data scientists need to use appropriate explainability techniques to explain the results obtained from the AI models. The XMANAI platform will provide data scientists with common explainability approaches to highlight the reasons and get insights on how the results occurred (e.g., relevance of inputs). According to the nature of the problem, they can decide which explainability technique fits best the task and create simple visualizations to effectively communicate the results to the business users. Thus, visualization libraries and tools to communicate, in a comprehensible way, the outputs to business users are necessary as well. It is also important to compare the explainability results of different approaches, in cases where it is not clear which explainability method should be followed.

By explaining the results of the AI models, data scientists can get insights about the models' inner workings and the way their decisions are taken, inspect for biases in the results and redesign the pipeline if needed. They are also able to avoid obvious outputs as meaningful explanations of the AI models. In addition, understanding the way the models function and produce the outputs increases the confidence for the accuracy and the robustness of the models. Business users are also benefited from the explanations and the visualizations produced, as the information they receive can help them take beneficial decisions for their business operations.

Despite the many benefits of the explainability techniques, data scientists meet a lot of challenges during their attempt to produce valuable explanations. Explainability is a relatively new term in the fields of machine learning and AI. Due to this fact, there are not a lot of available explainability libraries that are consistent and robust in their implementation. Most of the times it is rather difficult to break down the complexity of a black box model and present the explainable results in a useful and usable way for the business users (e.g., in the form of actions that need to be made). Moreover, matching the expectations of the business user is not easily guaranteed as the selection of the proper explainable method is rarely an easy task.

Table 2-7: Data Scientist Journey Phase 4 – AI Experimentation

Phase 4: AI Experimentation - Explain AI models / results	
Actions	Determine the explainability method that fits better to the problem by checking the needs of the business users
	Create visualizations to better convey the model results and the decisions to the business users
	Inspect for bias in the results and avoid obvious outputs as explanations
Expectations	Common explainability techniques to highlight the reasons for the obtained results (relevance of inputs, how decisions are taken, etc.)
	Visualization libraries appropriate to communicate the results to business users in a comprehensible way
	Compare the explainability results of different approaches
Benefits	Get insights about the way the models' decisions are taken (relevant features, biases) and redesign the pipeline if needed
	Increased confidence for the accuracy and robustness of the models.
	Present the results to business users in a way that is comprehensive to them and increases their trust in AI
Challenges	Difficulty in making explanations useful to the business user (e.g., in the form of rules)
	Selecting the explainability techniques that match the expectations of the business user per problem
	Breaking down the complexity of black box models
	Lack of available explainable AI libraries which are consistent and robust

Phase 5: AI Experimentation - Evaluate AI models / pipelines

During phase 5, data scientists monitor the models' performance using different metrics and validation methods. The XMANAI platform should provide them with the option to choose different performance metrics according to the model in use (e.g., different metrics for regression versus classification models), so that the right metrics will be used for the validation process. In addition, data scientists should be able to easily apply the chosen validation method and even incorporate it in

the AI pipeline (e.g., cross validation approach, hold out, k-fold cross validation, bootstraping). Common visualization libraries for visualizing the performance metrics of an experiment are also expected, as well as the ability to log experiments (model runs) & metrics and be able to compare them at will. It is critical in this phase to evaluate not only the accuracy of the results, but also the validity and understanding of the explanations for the target business users, by appropriately involving them.

The evaluation phase of AI models gives data scientists the opportunity to debug and improve the performance of the trained models. They can choose the most appropriate, explainable, reliable and accurate model (best model parameters), by monitoring and visualizing the performance metrics of different configurations and architectures. Data scientists can also recognize the most relevant features through the evaluation procedure and use them for a potential retraining in the future.

The challenges that are encountered at the model evaluation phase are mostly related to the choice of the appropriate evaluation method, in order to make a reasonable decision about the best model (in terms of explainability and performance). Having to find the right metrics to attach importance to is also significant, as for different applications and business users different metrics are more important than others and thus different models will be chosen as best.

Table 2-8: Data Scientist Journey Phase 5 – AI Experimentation

Phase 5: Evaluate AI models / pipelines	
Actions	Define the proper validation method according to the problem
	Choose the right performance metrics that will be used for the validation process
	Log the results of different experiments and compare the performance of the different models
	Create visualizations to efficiently present the validation results in a graphical format
Expectations	Select different metrics according to the model in use (e.g. to measure performance, vulnerability, explainability)
	Log experiments (model runs) & metrics in order to be able to select which experiments to compare
	Extract feature importance in order to see what features were most influencing for a decision
	Commonly used visualizations for the performance metrics of an experiment or a set of experiments
	Effortless application of validation methods incorporated even in the pipeline (e.g., cross validation approach)
Benefits	Debug and improve model performance
	Selection of the most reliable, accurate and trustful models (best parameters)
Challenges	Choose appropriate model evaluation techniques
	Finding the right metrics to attach importance to, in agreement with the needs of the business users

2.4 Data Engineer Journey

The Explainable AI Business User Journey describes how data engineers (within a manufacturer and in general) currently operate in their everyday work (as-is-situation) and what is the expected to-be situation with the XMANAI Platform.

2.4.1 AS-IS Situation

As in the case of data scientist, a data engineer is also a role that is hard to find in a manufacturing environment. In the XMANAI case, only one out of the four demonstrator partners has already employed data engineers that help in their day-to-day activities.

The data engineers' involvement focuses on extracting data related to AI projects from legacy systems and storing them in analytics-optimized databases. In general, data engineers need to ensure that data scientists do not build hard-coded pipelines that cannot be used in production settings and that the pipelines are portable and compatible with the available architectures and technologies. In addition, recognizing the edge cases of data pipelines and testing them before deployment is a demanding process that needs to be done, as well as ensuring data scalability at all stages of the pipeline.

Legacy relational databases and spreadsheets, such as MS SQL and Microsoft Excel, are still encountered as primary data sources, as in the case of FORD demonstrator. Even more so, in some particular cases, data recording for certain processes may be done manually, like in the cases of CNHi and UNIMETRIK, making this process more time-consuming and prone to human error.

In general, a manufacturing company hires a data engineer to organize, transform and manage data sources. This means to find and use tools to handle the vast amount of internal and external data available to the enterprise, to store all the available information securely and reliably, and to facilitate the ingestion and communication of real time sensor data (Ismail et al, 2019). The groundwork for them is to make data available as needed, thus, they have to design a data pipeline, which describes the workflow through which data originates from source and goes through a variety of processing steps to enable data analytics. Truth be told, in many situations this work is undertaken by the current IT department (even if they do not have a data engineer in the team) of the manufacturer or it is assigned to an external IT company.

2.4.2 To-BE Situation with XMANAI

The user journey of a data engineer consists of 3 phases: Phase 1 refers to the ingestion and handling of data in a collaborative way. Phase 2 concerns the collaborative design of AI in collaboration with data scientists and finally phase 3 deals with deploying the AI pipelines to production.

As all XAI User Journeys, the Data Engineer journey was created collaboratively by the XMANAI partners as displayed through the consolidated Miro boards in Figure 2-5.

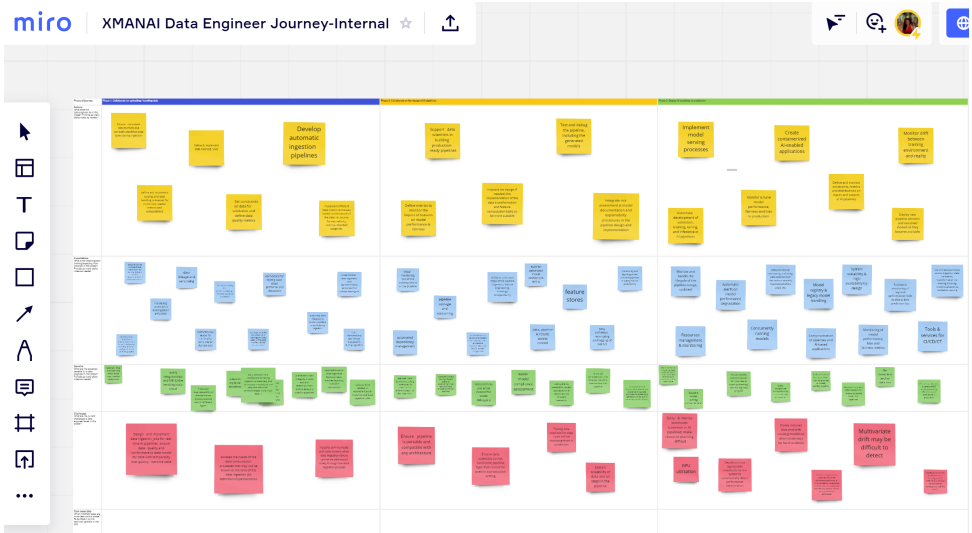


Figure 2-5: XAI Data Engineer Journey (consolidated Miro board)

Phase 1: AI Preparation - Collaborate on uploading / handling data

During phase 1, data engineers need to handle the data ingestion process. More specifically, the XMANAI platform should provide them with tools to design scalable and portable data ingestion jobs, that are easily configurable with transformation templates to handle various types of data. In this way, the engineers will be able to build data ingestion processes based on the nature of the data i.e., source, format, velocity, volume, intended usage. In addition, during ingestion it should be possible to map the data to an underlying data model in order to ensure they are understandable to any stakeholder that will utilize them. Moreover, ingestion processes should be monitored and controlled to facilitate data quality checks. Tracking data drift for continuous data ingestion processes could also be supported in order to monitor data quality. Other important features are the existence of flexible data storage, data lineage and versioning tools as well as connectors for the most used cloud platforms and databases. Data engineers should be able to get access to data as needed and keep track of the changes over time. XMANAI should also provide semantic data models for commonly encountered domain data and give the data engineers the ability to define data handling processes as needed.

In this context, data engineers expect to reduce the time needed for making the data ready for the AI pipelines (e.g., faster implementation of data transformation), as well as for monitoring, maintaining and scaling the implemented data ingestion jobs. In addition, data ingestion from common sources will be made faster, by using available connectors. Interoperability of datasets of different types and sources will be increased as well as the data quality. Moreover, data engineers will be able to early detect any failing ingestion processes and data drifting even before data are used in pipelines.

However, there are some challenges that complicate the processes of phase 1. Often, it is difficult for data engineers to foresee the needs of data consumption processes, that are unknown at the implementation time of data ingestion jobs. Another difficulty refers to designing data ingestion jobs for real-time AI pipelines. In this case, having data with temporality that quickly become stale, makes it difficult to build ingestion jobs that guarantee data quality and conformance to an appropriate data model. Finally in cases where data ingestion failures cannot be addressed solely by the data ingestion processes, data engineers should be able to communicate with data owners (e.g., business users) that can give some insights on the issues yet achieving this communication is most of the times challenging.

Table 2-9: Data Engineer Journey Phase 1 – AI Preparation

Phase 1: AI Preparation - Collaborate on uploading / handling data	
Actions	Develop automatic ingestion pipelines
	Ensure consistent data understanding during ingestion
	Define & implement data cleaning rules
	Define and implement queries and data handling processes for commonly needed metrics and computations
	Set constraints on data for validation and define data quality metrics
Expectations	Definition of as-needed data handling processes, i.e. not strictly upon data ingestion but configurable later on in the pipeline
	Define appropriate metadata for existing datasets and their columns/attributes/nodes/properties
	Customizable data ingestion and transformation templates for various data types
	Data mapping to an appropriate domain-specific data model during ingestion
	Monitoring and control over ingestion processes
	Data drift tracking for continuous/recurring data ingestion processes

	Data lineage and versioning
	Connectors for the most commonly used cloud platforms and databases
	By-design scalability and portability of data ingestion jobs
	Easily configurable and flexible data storage options
Benefits	Reduced time to make data ready to be consumed by AI pipelines
	Early detection of & notifications for failing ingestion process (e.g. due to increase in workload and required resources, errors/changes in the input data...)
	Easily integrate data and link to the (existing) data cloud
	Increased interoperability of datasets from diverse sources and of different types
	Reduced time to implement data transformations
	Increased and more easily assessed data quality based on conformity to a common model
	Automated data integrity checks and drift detection even before data are used in pipelines
	Reduced time for data ingestion from common sources by using available connectors
	Reduced time needed to maintain & scale implemented data ingestion jobs
Challenges	Design and implement data ingestion jobs for AI pipelines: ensure data quality and conformance to data model for data with temporality that quickly become stale
	Foresee the needs of the data consumption processes that may not be known at the time of the data ingestion job definition/implementation
	Need to communicate with data owners when data ingestion failure cannot be addressed solely through the data ingestion process

Phase 2: AI Experimentation - Collaborate on the design of AI pipelines

In this phase, data engineers need to make sure that designing AI pipelines will involve all the necessary parts for the procedure to be easily implemented, controlled and scaled. To that end, automated model packaging and testing are required as well as visual design of the pipeline. The platform should also guarantee the existence of standardised and reusable templates (e.g., for feature transformations and ML/DL). Each stage of the pipeline should be able to be scaled independently, so that data engineers can support data scientists in building AI pipelines easily and securely. In this direction, feature stores can be leveraged due to the reuse of features, while pipeline versioning, access control and experiment tracking can add to the collaboration with data scientists. Along these lines, assessing the relevance of features and parameters in model performance need to be facilitated.

The benefits that shall occur in this step are related mostly to the reduction of time needed for the various AI pipeline design. Specifically, less time will be required for writing "glue-code" to connect the different steps of the AI pipeline, for packaging and testing models across different scenarios and for debugging models. The ability to pre-compute the features and avoid duplicate computations will also accelerate results' generation and will optimize the use of resources. In addition, direct feedback loops among different teams, through a common interface will be facilitated while assessing models compliance adds value as well.

Data engineers, however, need to handle some challenges in this phase as well. They need to ensure that data scientists do not build hard-coded pipelines that cannot be used in production setting and that the pipelines are portable and compatible with all architectures. In addition, recognizing the edge cases of data pipelines and testing them before deployment is a demanding process that needs to be done, while ensuring data scalability at all stages of the pipeline.

Table 2-10: Data Engineer Journey Phase 1 – AI Experimentation

Phase 2: AI Experimentation - Collaborate on the design of AI pipelines	
Actions	Support data scientists in building production - ready pipelines
	Define metrics to monitor the impact of features on model performance & fairness
	Improve (re-design if needed) the implementation of the data transformation and feature computation tasks to be more scalable
	Test and debug the pipeline, including the generated models
	Integrate model risk assessment & model documentation and explainability procedures in the pipeline design and implementation
Expectations	Visual design of a pipeline
	Automated dependency management
	Pipeline storage and versioning
	Standardised and reusable templates for common tasks, e.g., feature transformations and ML/DL models' configuration
	Ability to scale each stage of the pipeline (ingestion, feature engineering, training) independently
	Automated model packaging, testing and security assessment
	Data, pipeline & results access control
	Feature stores
	Easy definition, monitoring and logging of metrics
	Modularity and layering of the available pipeline building blocks (templates)
Benefits	Reduced time devoted to writing "glue-code" to connect the different steps of the AI pipeline
	Elimination of silos among different teams by establishing direct feedback loops through a common interface
	Reduced time and effort needed for model debugging
	Easier model compliance assessment
	Reduced time needed for model packaging and testing across different scenarios
	Ensured consistency in the libraries' versions used across the pipeline
	Pre-computation of features when possible and avoidance of duplicate computations accelerating generation of results and optimising use of resources
Challenges	Ensure pipeline is portable and compatible with any architecture
	Ensure data scientists do not hard-code pipeline logic that cannot be used in a production setting
	Testing data pipelines for edge cases before deploying them in production
	Ensure scalability of data and all steps in the pipeline

Phase 3: AI Application - Deploy AI pipelines to production

Phase 3 includes all the processes needed for the AI pipeline to be efficiently deployed to production, in real-life manufacturing settings. Therefore, data engineers need to be able to monitor and handle the lifecycle of the pipeline (e.g., its usage, updates needed) and make sure that the system is scalable

by design. They need to deal with continuous integration, development and training (CI/CD/CT) activities to automate the validation, training and tuning of the pipelines and also job orchestration tools across all the processes of an AI pipeline (data ingestion, validation, transformation, re-training & tuning, model evaluation/validation and serving). In addition, monitoring model performance, bias and fairness metrics, as well as data and model versioning, are quite critical. The platform should also support a model registry so that engineers can handle legacy models. Moreover, they should be able to slice prediction logs for monitoring regional performance, so that they can monitor and tune model performance in production. Other important features are the resource management and monitoring, the containerization of pipelines and the ability to concurrently run models. Automatic alerts on model performance degradation shall contribute to monitoring data drifts that occur over time, so that engineers can deploy new pipeline versions and retrained models. Finally, engineers should handle access control on the model registry and the pipelines.

Data engineers expect to leverage a scalable model serving infrastructure, reduce the required time to transition from training and testing pipelines to production-ready model serving pipelines and reduce the effort to release updated models. In addition, drift and errors needs to be detected easily and early before they affect the model predictions in production in order to eliminate downtime as well as data loss. Another benefit is that any security vulnerabilities will be detected early and accessibility of information stored will be protected.

The challenges that appear at this phase are related to managing the resources available and handling models' performance in production. More specifically, managing GPU utilization and resource planning is difficult, due to spiky and intense workloads which are common in AI pipelines. In addition, feedback loop from prediction to re-training may be difficult to achieve or too slow to address the real-life needs. Re-training needs to occur after detection of performance deterioration, however deciding on the most appropriate performance thresholds, is not an easy task. The same applies for detecting multivariate drift, as well as slowly induced bias that causes models to deteriorate. Finally, it is difficult to ensure comprehensive observability of the developed pipeline by all involved teams, meaning that it is not easy to make the monitored metrics and the visualized result actionable for all the involved users.

Table 2-11: Data Engineer Journey Phase 3 – AI Application

Phase 3: AI Application - Deploy AI pipelines to production	
Actions	Create containerized AI-enabled applications
	Automate development of validation, training, tuning, and inference in AI pipelines
	Implement model serving processes
	Monitor & tune model performance, fairness and bias in production
	Define and monitor constraints, metrics and distributions on inputs and outputs of AI pipelines
	Monitor drift between training environment and reality
	Deploy new pipeline versions and retrained models as they become available
Expectations	Monitor and handle the lifecycle of the pipeline (usage, updates) through CI/CD/CT
	Resources management & monitoring
	Automatic alerts on model performance degradation
	Concurrently running models
	Data and model versioning, including data used to train the various models, hyperparameters used, etc.

	Model registry & legacy model handling
	Containerization of pipelines and AI-based applications
	System scalability & high availability by-design
	Monitoring of model performance, bias and fairness metrics
	Facilitated monitoring of regional performance: tools to slice & dice prediction logs
	Job orchestration across ingestion, data validation, transformation, re-training & tuning, model evaluation & validation, serving
Benefits	Visual monitoring of pipeline usage and performance reducing required time to ensure smooth operation or detect issues
	Scalable model serving infrastructure
	Ensure system responsiveness for real-time & batch data processing, as well as for big data
	Cyber compliance: mechanisms to detect security weaknesses
	Reduced time to transition from training pipeline to model serving pipeline
	Reduced time and effort needed to release updated models & pipelines
	No downtime and no data loss
	Ensure data integrity & detect drift & errors before they affect the model predictions in production
Challenges	Spiky & intense workloads (common in AI pipelines) make resource planning difficult
	GPU utilisation
	Decide on most appropriate thresholds for the system to automatically detect performance deterioration
	Slowly induced bias and drift causing models to deteriorate may be hard to detect
	Ensure comprehensive observability of the developed pipeline by all involved teams - ensure the metrics that are monitored and the way various results are visualised are actionable
	Feedback loop from prediction to re-training may be difficult to achieve or too slow to address the real-life needs

3 Technical Requirements

This section introduces the technical requirements that have emerged through brainstorming and focus groups by the XMANAI partners.

3.1 Overview

A requirement is a service, function or feature that a user needs in the software. Requirements can be functions, constraints, business rules or other elements that must be present to meet the need of the intended users. Requirement gathering techniques in agile software development come in many shapes and forms, but the most common form is a User Story. A User Story is a requirement expressed from the perspective of an end-user goal. User stories represent the needs of the customer in a simply written narrative that can be easily understood.

The main difference between user stories and use cases (as another common technique of requirement elicitation) is their objectives. The user story focuses on the experience and what the person using the product wants to be able to do. A traditional use case focuses on functionality and what the product should do.

One of the principles behind User Stories is that the product could be fully represented through the needs of its users. Because the User stories are short and simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system. The focus is on why and how the user interacts with the software. A user story is essentially a high-level definition of what the software should be capable of doing.

User story descriptions typically follow a simple template as a Card:

```
As a <role>, I want <goal> so that <benefit>
```

Bill Wake (2003) proposed the INVEST acronym that expresses the six characteristics which a proper user story should have.

- *Independent* – One user story should be independent of another (as much as possible). Dependencies between stories make planning, prioritization, and estimation much more difficult.
- *Negotiable* – Details of the story can be worked out during an Iteration planning meeting. A story with too much detail can limit conversations (at times).
- *Valuable* – Value to the customer needs to be evident.
- *Estimable* – There needs to be enough detail for the developers to estimate a user story to allow prioritization and planning of the story.
- *Small* – A good story should be small in effort, typically no more than 2-3 person weeks of effort.
- *Testable* – User stories should be testable with certain acceptance criteria.

To ensure that the XMANAI Platform meets the requirements of the different stakeholders (described in Section 2.1), an agile methodology is implemented for requirements elicitation, promoting interactive sessions with the XMANAI partners. Requirements of the XMANAI platform are extracted in the form of user stories, collaboratively by technical partners and business stakeholders through interactive sessions. XMANAI has been using Miro (www.miro.com), which is an online collaborative whiteboard platform, configured with separate User Story boards for each of the roles in XMANAI, namely Data Scientist, Data Engineer and Business Expert.

The interaction between the partners to fill-in the User Story boards has stimulated creativity and provided the opportunity to examine the needs not only from a technical point of view but also from a practical point of view, defining the behavior of the XMANAI platform towards the initial goal of achieving explainability in AI for manufacturing purposes. Several user stories have been extracted at the end of the brainstorming sessions. Later on, the extracted user stories were tagged and categorized by technical WPs of the XMANAI project, namely WP2, WP3 and WP4, and their corresponding tasks in order to be consolidated in the XMANAI requirements backlog.

Figure 3-1 represents the User Story board that was created collaboratively by the XMANAI partners for Data Scientists.

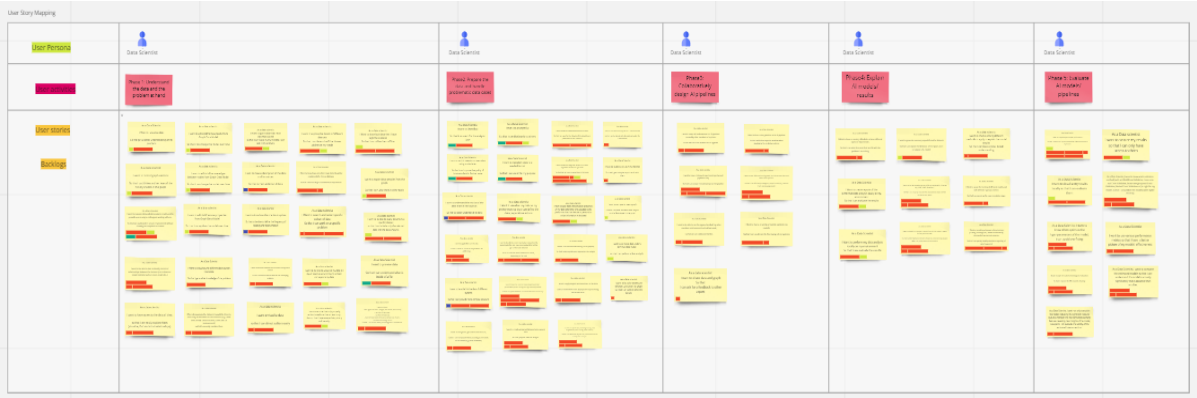


Figure 3-1: Miro User story board for Data Scientist

Figure 3-2 presents the User Story board that was created collaboratively by the XMANAI partners for Data Engineers.

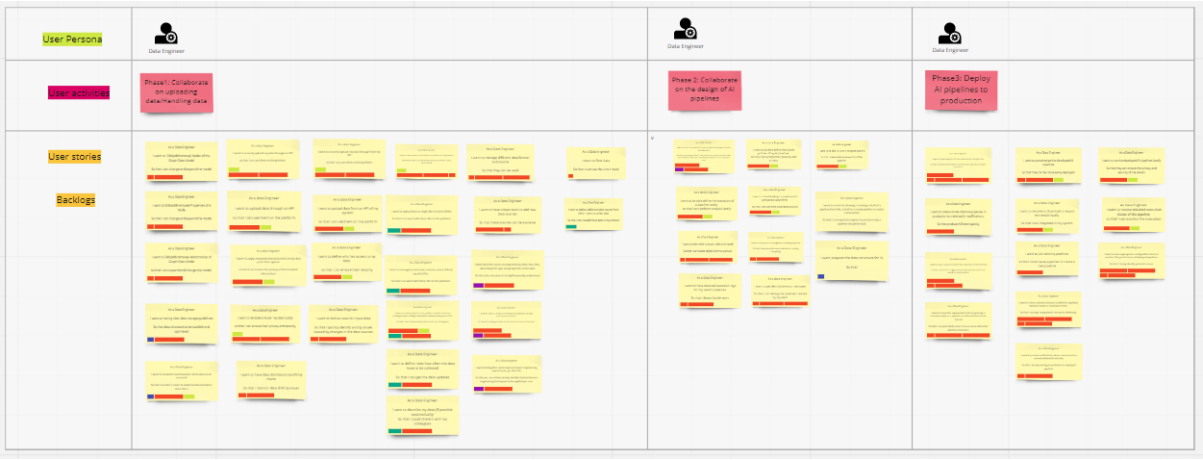


Figure 3-2: Miro User story board for Data Engineer

Figure 3-3 presents the User Story board that was created collaboratively by the XMANAI partners for Business Experts.

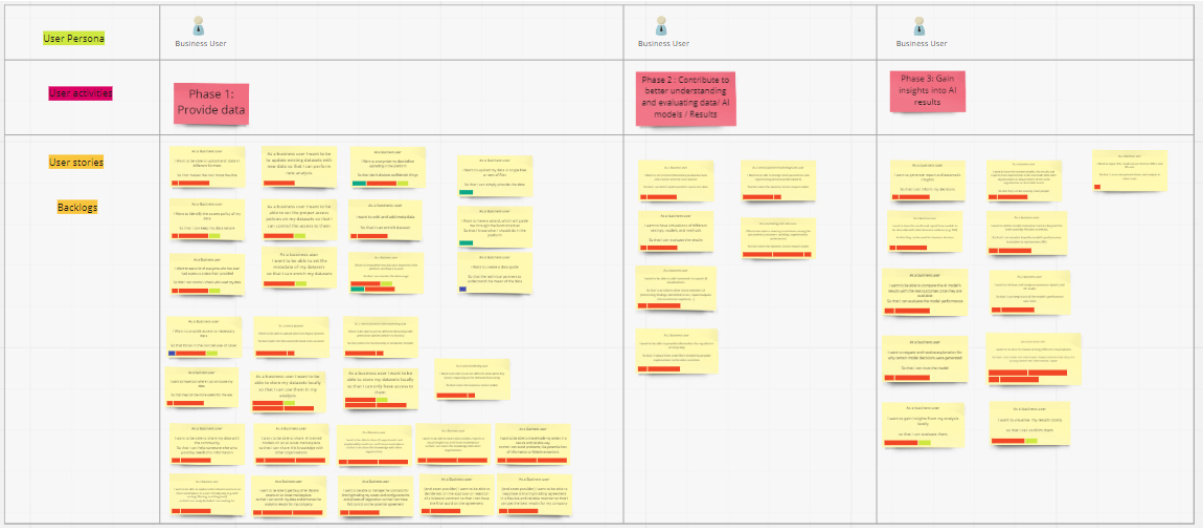


Figure 3-3: Miro User story board for Business User

3.2 Backlog

A product backlog is a list of the features and tasks that should be implemented as part of the project development. The product backlog can be presented as a list of user stories that should be performed by the development team to develop the desired product. User stories are the smallest unit of work in an agile framework that describes the requirements at a level of detail and fits perfectly in the product backlog. Indeed, the User story is the single source of work undertaken by an agile development team.

This section focuses on the backlog of the XMANAI platform and provides a detailed presentation of all technical requirements in the form of user stories. The XMANAI platform backlog is extracted collaboratively through brainstorming by partners through Miro. After several interactive sessions and further discussion, the XMANAI technical requirements are extracted and ready for prioritization as part of the MVP activities. Table 3-1 provides the list of the initial technical requirements.

Table 3-1: XMANAI Technical Requirements

No	Task No	As a…	I want to …	So that…
TR_1	T2.1	Data Scientist, Data Engineer, Business User	add/edit/remove new data sources for data/metadata import	Data/metadata from these sources can be available for analysis
TR_2	T2.1	Data Scientist, Data Engineer, Business User	define rules how often the data have to be collected	I can get regular data updates from an external source
TR_3	T2.1	Data Scientist, Data Engineer, Business User	upload data as single file or batch of files	I can work with these data in the platform
TR_4	T2.1	Data Engineer	Delete data and its associated metadata	I can remove data, which I don't need anymore
TR_5	T2.1 & T3.1	Admin, Data Engineer	create / import a data model for a specific domain	relevant data can be mapped to it and become available in the platform
TR_6	T2.1 & T3.1	Data Scientist, Business User	search for concepts, fields and relationships to an existing data model	I can ensure my specific data needs are addressed

No	Task No	As a…	I want to …	So that…
TR_7	T2.1 & T3.1	Admin, Data Engineer	manually add new concepts, fields and relationships to an existing data model	I can ensure the needs of data scientists and business users are addressed
TR_8	T2.1 & T3.1	Admin, Data Engineer	manually update the concepts, fields and relationships of an existing data model	I can change the model over time
TR_9	T2.1 & T3.1	Admin, Data Engineer	manually control the different versions of the data model	I can view and retrieve all the history of edits
TR_10	T2.1 & T3.1	Admin, Data Scientist	view the different data models	I can use an appropriate model per domain
TR_11	T2.1 & T3.1	Data Scientist	generate different representations (e.g. graph) of the data model	I can make it available to other applications / components
TR_12	T2.1	Data Scientist, Data Engineer, Business User	add/edit metadata for my data (based on a common metadata model)	I can improve the quality of data for further reuse
TR_13	T2.1	Data Scientist, Data Engineer, Business User	define how my data are mapped to a data model	the data types and semantics of my data can be available to anyone who uses my data
TR_14	T2.1	Data Scientist, Data Engineer, Business User	define whether and how my data should be cleaned	I can increase their quality before they are stored
TR_15	T2.1	Data Scientist, Data Engineer, Business User	manage (create new, rename, move, delete) my datasets	I can work with them in the platform
TR_16	T2.1	Business user, Data Scientist, Data Engineer	update existing datasets with new data	I can perform a new analysis
TR_17	T2.1 & T2.3	Data Scientist	download data that I have legitimate access as a file	I can use them offline
TR_18	T2.1 & T2.3, T5.X	Data Scientist	retrieve data that I have legitimate access through an API	I can use them in the XMANAI manufacturing apps
TR_19	T2.1 & T2.3, T3.2, T3.3	Data Scientist, Data Engineer	export data samples from the dataset	I can view them in other XMANAI components and external tools
TR_20	T2.1, T2.2, T3.5	Business user, Data Engineer	be able to store my datasets on my premises	I can only have access to them and I can use them in my analysis
TR_21	T2.2	Business user, Data Scientist	able to define the proper access policies on my assets (datasets, AI models, AI pipelines, experiments, analysis results)	I can define who shall have access to my assets and under which circumstances
TR_22	T2.2	Business user, Data Scientist	able to combine multiple access policies on my assets (datasets, AI models, AI pipelines, experiments, analysis results)	I can define more complex access restrictions to my assets
TR_23	T2.2	Business user, Data Scientist	able to define access policies based on various attributes of the requestor or a specific asset (datasets, AI models, AI pipelines, experiments, analysis results)	I can define who shall have access to my assets and under which circumstances
TR_24	T2.2	Business user, Data Scientist	able to update or remove the access policies on my assets (datasets, AI models, AI pipelines, experiments, analysis results)	I can reconsider who shall have access to my assets

No	Task No	As a…	I want to …	So that…
TR_25	T2.2	Business user, Data Scientist	able to define the access level of my assets only to my organisation	I can provide access only to my organisation's users
TR_26	T2.2	Business user, Data Scientist	able to define the access level of my assets only to selected users outside my organisation	I can get support from other data scientists
TR_27	T2.2	Business user, Data Scientist	able enforce the access control decision based on my access policies	I can ensure that the access to my assets is always safeguarded
TR_28	T2.2	Data Scientist, Data Engineer, Business User	ensure that only properly authenticated users have access to my assets (datasets, AI models, AI pipelines, experiments, analysis results)	I can ensure their privacy and security
TR_29	T2.2	Data Scientist, Data Engineer, Business User	configure and apply different data anonymisation processes on my data before they are uploaded in the platform	I can assure no personal or critical information is disclosed
TR_30	T2.2	Data Scientist, Data Engineer, Business User	configure and apply different data anonymisation processes on samples of data before they are uploaded in the platform	I can understand what changes are to be performed on my data
TR_31	T2.2, T5.x	Business user, Data Scientist	ensure that my data are transferred between the different layers of the platform securely	I can ensure that my data will not be disclosed to unauthorized parties
TR_32	T2.3	Business User, Data Scientist, Data Engineer	share my assets (e.g. datasets, AI models, features, AI pipelines, experiments, analysis results) with other organisations / users of my preference	I can help someone else who possibly needs this information
TR_33	T2.3	Business User	trade my assets (e.g. datasets, AI models, features, AI pipelines, experiments, analysis results) in a secure and reliable way for a specific time period	I can gain new revenues from my assets
TR_34	T2.3	Business User, Data Scientist, Data Engineer	search and explore other data/AI assets on an in a user-friendly way (e.g. based on metadata, with sorting, filtering, matching level)	I can easily find what I am looking for
TR_35	T2.3	Data Scientist, Data Engineer	view metadata of the selected asset (e.g. datasets, AI models, AI pipelines, analysis results)	I can determine if the asset addresses my needs
TR_36	T2.3 & T2.4	Business User, Data Scientist, Data Engineer	know which are the IPR holders involved in the asset I am interested in and what are their associated rights/licenses	I can take an informed decision for the asset acquisition
TR_37	T2.3	Business User, Data Scientist, Data Engineer	get access to assets (e.g. datasets, AI models, features, AI pipelines, experiments, analysis results) created by other users, in a secure and reliable way for a specific time period	I can enrich my data and enhance the analytics results for my company
TR_38	T2.3	Business User, Data Scientist, Data Engineer	buy assets (e.g. datasets, AI models, features, AI pipelines, experiments, analysis results) created by other users, in a secure and reliable way for a specific time period	I can enrich my data and enhance the analytics results for my company
TR_39	T2.3	Business User, Data Scientist, Data Engineer	manage the contracts for sharing/trading my assets and configure terms	I can have full control on the potential agreement
TR_40	T2.3	Business user, Data Scientist, Data	be able to negotiate a sharing/trading agreement in a flexible and reliable manner	I can achieve the best outcome for my company

No	Task No	As a…	I want to …	So that…
		Engineer (and asset provider)		
TR_41	T2.1 T2.3, T3.2	Data Scientist	consistently handle missing data in my data or by finding other relevant datasets	I can anticipate data drift
TR_42	T2.4	Data Scientist, Data Engineer	view how often the data is updated and when	I can know if I have the latest data or change the update frequency if needed
TR_43	T2.4	Data Scientist	have a control version of the datasets	I am sure that I use the latest updated data
TR_44	T2.4 & T3.3, T4.1, T4.2	Data Scientist, Data Engineer	have a control version of assets (AI models, AI pipelines, features, experiments, results)	I can keep track of the changes introduced and limit the impact of changes on existing pipelines
TR_45	T2.4	Business user	XMANAI to register each access event (based on actions performed) of other users to my data or other assets	I can have detailed logs who and when accessed my data and other assets
TR_46	T2.2 & T2.4	Business user	see a list of users who has ever had access to an asset that I provided and what activities were performed	I can monitor my assets usage
TR_47	T2.4	Business user	view which data and assets I have shared with whom	I can monitor my sharing activities
TR_48	T2.2 & T2.4	Business user	XMANAI to register to whom I permitted access to my data or other assets	I can log the sharing my data and other assets with other users
TR_49	T2.4	Data Scientist	check the IPR of the assets involved in an AI pipeline	all assets included in a pipeline are used in alignment with their licenses
TR_50	T3.2	Data Scientist	query data to which I have legitimate access	I can find a subset of the data I can use in my analysis
TR_51	T2.1 & T3.2	Data Scientist	know the data types and the semantics per field that appears in the data	I can quickly understand the data that I will use in an analysis
TR_52	T3.2	Data Scientist	preview a sample of the data	I can obtain a more concrete understanding of the data at hand
TR_53	T3.2	Data Scientist	view data distribution/profiling charts or summary statistics for the data (e.g. number of missing values, min/max values)	I can monitor data drifting issues
TR_54	T3.2	Data Scientist	define rules for input data	I quickly identify wrong values caused by changes in the data sources
TR_55	T3.2	Data Scientist	create new features based on the current data (like min, max, mean values) that will be part of the same dataset	I can have more informative datasets, depending on the task
TR_56	T3.2	Data Scientist	handle missing values (impute)	I can prepare the data for the subsequent AI analysis
TR_57	T3.2	Data Scientist	encode categorical data	I can prepare the data for the subsequent AI analysis
TR_58	T3.2	Data Scientist	apply scaling and data normalization	I can prepare the data for the subsequent AI analysis
TR_59	T3.2	Data Scientist	easily split the data for training and evaluation (data segmentation)	I can train and apply the models as I see fit
TR_60	T3.2	Data Scientist	apply simple transformations on the data	I make them more appropriate for processing and visualisation

No	Task No	As a…	I want to …	So that…
TR_61	T3.2	Data Scientist	change the data type of some features	I can manipulate them according to the needs of an AI model (e.g. convert to datetime)
TR_62	T3.2	Data Scientist	apply data augmentation techniques	I can reduce overfitting of the models
TR_63	T3.2	Business User	perform calculations over my data	I can keep track of important Key Performance Indicators that are important for my business
TR_64	T3.2 (& T3.1)	Business User	be able to provide information for my data in an easy way	I reduce time and effort needed to provide explanations to the data scientists
TR_65	T3.3	Data Scientist, Data Engineer	define and configure an AI pipeline for training, testing and/or production purposes	I can provide a solution for a specific problem
TR_66	T3.3 (& T4.1 / T4.2)	Data Scientist, Data Engineer	include compatible baseline algorithms in an AI pipeline	I can provide a solution for a specific problem
TR_67	T3.3 (& T4.1 / T4.2)	Data Scientist, Data Engineer	include compatible trained models in an AI pipeline	I can provide a solution for a specific problem
TR_68	T3.3	Data Scientist	register a trained AI model I have created as part of an AI pipeline	I can reuse it in my AI pipelines
TR_69	T3.3, T2.3	Data Scientist, Data Engineer	collaborate in the configuration of AI pipelines with selected users (within my organization or external to my organisation)	I can create optimal workflows for a specific problem
TR_70	T3.3	Data Scientist, Data Engineer	define pipelines that can be used as templates for specific problems	my colleagues and I can reuse them
TR_71	T3.3	Data Scientist, Data Engineer	clone a designed pipeline	I can create alternative version of the pipeline and improve its performance without re-creating it from scratch
TR_72	T3.3	Data Scientist, Data Engineer	join designed pipelines	I can create advanced combinations (e.g. training pipeline with testing pipeline, multiple training pipelines)
TR_73	T3.3	Data Scientist, Data Engineer	execute step-by-step an AI pipeline over sample data	I can ensure that the result is the intended one
TR_74	T3.3	Data Scientist, Data Engineer	reuse common features in different pipelines	I do not need to recompute them
TR_75	T3.3 (& T4.1 & T4.2)	Data Scientist	configure training to control parameters, such as learning rate reduction when a metric has stopped improving or stop it	I can notified if there is any problem during training
TR_76	T3.3	Data Scientist	receive recommendations for automated feature selection	I can perform feature engineering in a faster / easier manner, in cases of high-dimensionality data
TR_77	T3.3	Data Scientist	choose among different methods to apply for explaining an AI pipeline (including its input data, models and results)	I can select the ones that best fit with the problem I am solving
TR_78	T3.3	Data Scientist	properly adjust the explanations depending on the user profile	I can increase understanding of the results for its intended users

No	Task No	As a…	I want to …	So that…
TR_79	T3.3	Data Scientist	define summary statistics to be computed for an AI pipeline or part of it	I can explain the behavior of the inputs and / or outputs of a pipeline / model
TR_80	T3.3	Data Scientist	add annotations and comments in AI pipelines	I can better explain the results
TR_81	T3.3 (T3.4 & T3.5)	Data Scientist	define whether the analysis results should be saved as a new dataset or update an existing one	I can re-use the analysis results
TR_82	T3.3 (T3.4 & T3.5)	Data Scientist, Data Engineer	export the results of an analysis (AI pipeline)	I can create presentations and analysis in other tools (e.g. Office, BI tools)
TR_83	T3.3	Data Scientist	add comments to visualisations of AI pipelines	I can inform other team members of [interesting findings, identified errors, inputs/outputs that need to be explored,…]
TR_84	T3.3	Business User	properly visualise the explanations depending on the user profile	I can adapt the information I receive, according to my needs
TR_85	T3.3	Data Scientist	define the parameters and metrics of the experiments associated with an AI pipeline	I can track and compare my experiments
TR_86	T3.3 (& T4.1 / T4.2)	Data Scientist	run automatic tests on the registered AI models (when including them in an AI pipeline)	I can quickly check that the trained models are robust and fault-tolerant
TR_87	T3.3 (& T4.3)	Data Scientist	support the inclusion of different performance metrics as part of an AI pipeline	I can obtain a better picture of my pipelines' / models' effectiveness according to my needs
TR_88	T3.3	Data Scientist	generate multiple visualisations as output of an AI pipeline	I can make it available to the involved users
TR_89	T3.3	Business User	view a visualisation including the results and their explanations	I can take informed decisions
TR_90	T3.3	Business User	choose among different visualisations	I can create the charts and measurements that help me quickly detect the information I want
TR_91	T3.3	Business User	request explanations for why certain predictions were generated in an AI pipeline	I can trust the model
TR_92	T3.3	Data Scientist	respond to requests for explanations of an AI pipeline generated by users	I can help the business users to trust the model
TR_93	T3.3 (& T2.3)	Data Scientist	have a common metadata model for describing my AI pipelines, experiments, results	I can share them with other users
TR_94	T3.3 (& T4.3)	Business User	define model evaluation metrics (beyond the ones used by the data scientists)	I can monitor how the model's performance translates to my business KPIs
TR_95	T3.3 (T4.3)	Business User	retrieve and compare previous AI model's results with the real outcomes once they are available	I can keep track of the model's performance over time
TR_96	T3.3	Data Scientist, Data Engineer	define where my AI pipeline will be executed	I can ensure the analysis is securely executed in infrastructures under my control
TR_97	T3.3, T3.4 & T3.5	Data Scientist	track the performance of my trainings (regarding stability and converge of the results, execution times, the use of computational resources, etc.)	I can be automatically informed of non-normal scenarios via descriptive error messages

No	Task No	As a…	I want to …	So that…
TR_98	T3.3, T3.4 & T3.5	Data Scientist	implement different mechanisms to save a checkpoint of the AI model	I can resume the training of a model from the previous point
TR_99	T3.3, T3.4 & T3.5	Data Scientist	know when the hyperparameters of the model are optimized	I can avoid overfitting
TR_100	T3.3, T3.4 & T3.5	Data Scientist	launch and queue the training of different models and with different hyperparameters	I can speed up the experimentation stage
TR_101	T3.4 & T3.5	Data Scientist, Data Engineer	get descriptive error messages	I can debug the AI pipelines that have been executed
TR_102	T3.4 & T3.5	Data Scientist, Data Engineer	view detailed execution logs for an AI pipeline	I can detect bottlenecks once my AI pipeline has been executed
TR_103	T3.4 & T3.5	Data Scientist, Data Engineer	keep a history of experiments performed (pipeline runs)	I can compare results & improve the pipeline
TR_104	T3.4 & T3.5	Data engineer	schedule the execution of the AI pipelines	I can have at my disposal up-to-date results
TR_105	T3.4 & T3.5	Business User, Data Scientist	store the execution results	I can retrieve them to use them in external systems
TR_106	T3.4 & T3.5	Data Engineer	receive notifications when certain metrics exceed defined thresholds	I can timely investigate problems in deployed pipeline
TR_107	T3.4	Data Scientist, Data Engineer	have the required resources and automatic parallelization, when dealing with big data manipulation	I can execute an AI pipeline in a faster and more reliable way
TR_108	T3.5	Business user, Data Scientist	execute AI pipelines locally on my private cloud or servers on premise	I can perform my analysis in a secure and trusted environment
TR_109	T3.5	Data Scientist	create reports of the performed data analysis locally on my environment	I can evaluate the results
TR_110	T3.4 & T3.5	Data Scientist	export my analysis results or AI models as files	I can import them on my organisation's systems
TR_111	T3.4 & T3.5	Data Scientist	retrieve my analysis results through an API	I can use them in the XMANAI manufacturing apps
TR_112	T3.4 & T3.5	Data Engineer	containerize the developed AI pipelines	they can be more easily deployed
TR_113	T3.5	Data Scientist	able to setup a data analysis execution and results visualization environment easily on private cloud or servers on premise	I can leverage the execution of the analysis and visualization of results on infrastructures I control
TR_114	T4.1 & T4.2	Data Scientist	provide wrappers (register) for baseline algorithms from selected ML libraries (e.g. sk-learn, spark mllib)	I can include them in my AI pipeline
TR_115	T4.1 & T4.2	Data Scientist	provide wrappers (register) for baseline algorithms for clustering, classification, regression, dimensionality reduction	I can include them in my AI pipeline
TR_116	T4.1 & T4.2	Data Scientist	provide wrappers (register) for baseline algorithms from selected DL libraries (e.g. tensorflow)	I can include them in my AI pipeline
TR_117	T4.1 & T4.2	Data Scientist	provide wrappers (register) for explainability techniques	I can include them in my AI pipeline
TR_118	T4.1 & T4.2 (& T3.3, T2.3)	Data Scientist	have a common metadata model for describing my trained models (hyperparameters, parameters, code, metrics...)	I can share them with other users
TR_119	T4.1 & T4.2 (&	Data Scientist	package the trained AI models I want to register (following specific	I can upload them to the XMANAI catalogue

No	Task No	As a…	I want to …	So that…
	T3.3, T2.3)		guidelines for directory tree, programming language, name of files, etc.)	
TR_120	T4.1, T4.2 & T3.3	Data Scientist	register a trained AI model I have created offline to solve a specific problem	I can make it available to be reused by other users
TR_121	T4.1, T4.2 & T3.3, T2.4	Data Scientist	keep versioning for the trained AI model I have created offline	I can retrain the model without affecting all AI pipelines it has been reused
TR_122	T4.1, T4.2 & T3.3	Data scientist	follow specific guidelines for the explainability techniques (e.g. surrogate models) I want to register	all techniques and their associated metadata (including python packages requirements) can be packaged and available in the AI pipelines
TR_123	T4.1, T4.2 & T3.3	Data Scientist	register a surrogate model I have created offline	I can reuse it in my AI pipelines
TR_124	T4.3	Data Scientist	know which validation method is right for my AI model	I can validate the model(s) with higher accuracy
TR_125	T4.3 (T4.1, T4.2, T3.3)	Data Scientist	use various performance metrics for the AI models	I have a better picture of my models' effectiveness
TR_126	T4.3 (T3.3)	Data Scientist	generate an evaluation report for each trained ML/DL model	users that insert it in their AI pipelines are aware of its performance
TR_127	T4.3	Business User (Expert)	evaluate the validity of the explanations and provide feedback	I can improve the AI models and AI pipelines to solve a specific problem
TR_128	T4.3 (T3.3)	Data Scientist	compare results from different models created for a particular task	I can gain an understanding of which factors aid in the performance of the models (which features helped, how the different preprocessing steps affected the result
TR_129	T4.3 (T3.3)	Data Scientist	compare performances of different AI pipelines	I can track the improvements of the different versions of my pipelines / models
TR_130	T4.3	Business User	view reports of the experiments (simulations of different settings, models, and methods) to solve a specific problem	I can evaluate the results
TR_131	T4.4	Data Scientist	test the training data sets at each step to detect possible poisoned data points	I can check the integrity of data sets
TR_132	T4.4	Data Scientist	filter poisoned data points out and retrain the model	I can repair possible poisoning of the model
TR_133	T4.4	Data Scientist, Data Engineer	generate adversarial examples	I can create a robust model against adversarial attacks
TR_134	T4.4	Data Scientist	check training data sets for possible unfair biases	I can prevent possible discriminatory biases of the model

It needs to be noted that all technical requirements for this deliverable refer to the XMANAI Platform developed through the activities of WP2-WP5, and not to the manufacturing apps that shall be developed in WP6 to address the specific manufacturing cases of the XMANAI Demonstrators.

3.3 Technical Requirements vs Business Requirements

A correlation between the technical requirements and the business requirements (from D6.1) to highlight how the business requirements have been considered. It needs to be noted that there might be multiple technical requirements relevant for each business requirement.

Table 3-2: Business – Technical Requirements Alignment

Business Requirements		Technical Requirements No.
No.	**Description**	
BR_1	When a failure affect occurs, XMANAI shall represent in real time the machine/part related to the failure.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_97, TR_114, TR_115, TR_116
BR_2	XMANAI shall send advice based on real time data to help to take actions about the stocks.	TR_17, TR_18, TR_19, TR_65, TR_66, TR_67, TR_68
BR_3	XMANAI should integrate all corporative data to know the status of machines in real time.	TR_1, TR_3, TR_16, TR_42
BR_4	XMANAI should allow operators and engineers to visualize the data from any locations and from multiple devices (smartphone, laptops).	TR_20, TR_51, TR_52, TR_53, TR_96, TR_98, TR_108, TR_113
BR_5	XMANAI should save historical data to help engineers to review the historical actions.	TR_45, TR_103
BR_6	XMANAI shall show direct and visual alarms to alert about critical situations	TR_73, TR_76, TR_97, TR_101, TR_106
BR_7	XMANAI should allow an operator to understand the root causes in every working situation.	TR_75, TR_95, TR_97, TR_127
BR_8	XMANAI should have flexibility to be applied in different lines.	TR_65, TR_66, TR_67, TR_68, TR_93, TR_118, TR_123
BR_9	XMANAI should provide advices in terms of production plan.	TR_65, TR_66, TR_67, TR_68, TR_84, TR_89, TR_91, TR_92, TR_117, TR_127
BR_10	XMANAI should read and integrate data from corporate databases and external sources.	TR_1, TR_3, TR_16, TR_32, TR_42, TR_43
BR_11	XMANAI should act as simulator of different planning scenarios.	TR_65, TR_66, TR_67, TR_68, TR_123, TR_128, TR_129, TR_130
BR_12	XMANAI should consider the production that is currently in the line but it hasn't yet finished.	TR_1, TR_2, TR_16
BR_13	XMANAI should be agile to replan when an unexpected event occurs in the line.	TR_65, TR_66, TR_67, TR_68, TR_77, TR_97, TR_100, TR_106,
BR_14	XMANAI should provide measurements of the deviation between the predicted plan and the real production.	TR_87, TR_94, TR_124, TR_125, TR_126, TR_127
BR_15	XMANAI should alert for critical parts, that sufficient stocks of some parts to finish the plan are not available.	TR_97, TR_101, TR_106
BR_16	XMANAI should allow a central planner to have a correct forecasting of D2C sales per day/product in a horizon of 3 months.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_85, TR_114, TR_115, TR_116, TR_120, TR_130
BR_17	XMANAI should generate demand forecasts on a daily basis.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_85, TR_114, TR_115, TR_116, TR_120, TR_130

Business Requirements		Technical Requirements No.
No.	Description	
BR_18	XMANAI shall allow central planners/D2C marketing, D2C sales/D2C logistics to visualise the key factors effects influencing demand profile.	TR_65, TR_66, TR_67, TR_68, TR_88, TR_89, TR_90, TR_114, TR_115, TR_116, TR_117
BR_19	XMANAI shall allow central planners/D2C marketing, D2C sales/D2C logistics to see the clustering of customer behavior.	TR_66, TR_88, TR_89, TR_90, TR_115, TR_116
BR_20	XMANAI shall allow central planners/D2C marketing, D2C sales/D2C logistics to visualise buying patterns per customer profile/product/period.	TR_65, TR_66, TR_67, TR_68, TR_78, TR_84, TR_88, TR_89, TR_90, TR_114, TR_115, TR_116, TR_117
BR_21	XMANAI shall allow D2C marketing/D2C sales to receive recommendations/input for promotional actions.	TR_65, TR_66, TR_67, TR_68, TR_78, TR_84, TR_114, TR_115, TR_116
BR_22	XMANAI shall allow central planners/D2C marketing, D2C sales/D2C logistics to simulate demand forecasting forcing the change in one or more key parameters.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_75, TR_79, TR_85, TR_114, TR_115, TR_116, TR_130
BR_23	XMANAI shall strictly authorize users for system access.	TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28, TR_31, TR_45, TR_46, TR_57, TR_58
BR_24	XMANAI shall strictly protect all sales data through encryption and secure data management, in compliance with Whirlpool data security policies.	TR_20, TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28, TR_31, TR_46, TR_57, TR_58, TR_108
BR_25	XMANAI shall provide full customer data anonymization	TR_29, TR_30, TR_57, TR_58
BR_26	XMANAI shall fully respect GDPR (General Data Protection Regulation) by a privacy by design approach, in compliance with Whirlpool data security policies.	TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28, TR_29, TR_30, TR_31, TR_45, TR_57, TR_58,
BR_27	XMANAI should provide an interactable digital twin able to forecast the behaviour of the machinery.	TR_65, TR_66, TR_67, TR_68, TR_69, TR_70, TR_114, TR_115, TR_116, TR_117, TR_120, TR_130
BR_28	XMANAI should provide an interactable digital twin able to improve the collaboration and communication between production parts and engineering CAD model of the parts themselves.	TR_64, TR_69, TR_80, TR_83, TR_91, TR_92
BR_29	XMANAI should provide alarms with description of the problem and visibility on how the problem has been forecasted.	TR_83, TR_84, TR_88, TR_89, TR_90, TR_97, TR_101, TR_109
BR_30	XMANAI should provide an interactable HMI able to improve the comprehension of the suggestion provided by the XAI and to navigate them.	TR_64, TR_77, TR_78, TR_84, TR_89, TR_91, TR_92, TR_93, TR_118, TR_127
BR_31	XMANAI should provide a set of data to the user identifying the problem before the critical moment, the maintenance/troubleshooting procedure to be executed and the parameters to be monitored during production.	TR_17, TR_18, TR_19, TR_75, TR_97, TR_106, TR_110, TR_111
BR_32	XMANAI should support the Blue Collar Worker is doing the maintenance/troubleshooting procedures with AR/XAI connection.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_77, TR_78, TR_89, TR_91, TR_92, TR_114, TR_115, TR_116, TR_117, TR_127
BR_33	XMANAI should learn from PMS past quality problems and parameters which was leading to the problem.	TR_65, TR_66, TR_67, TR_68, TR_70, TR_97, TR_114, TR_115, TR_116
BR_34	XMANAI should provide product quality risks starting from production parameters including explanation of why certain conditions happened or are forecasted to happen.	TR_77, TR_82, TR_88, TR_89, TR_90, TR_91, TR_92 TR_109, TR_125, TR_126, TR_127, TR_130

Business Requirements		Technical Requirements No.
No.	Description	
BR_35	XMANAI should generate a report with the quality risks for the production manager.	TR_82, TR_88, TR_89, TR_90, TR_109, TR_125, TR_126, TR_127, TR_130
BR_36	XMANAI shall be able to adjust the displayed criteria based on the geometry type and allows full control to add various GD&T checks and other specific location information for an element.	TR_53, TR_75, TR_79, TR_85, TR_94
BR_37	XMANAI shall be able to automatically add and connect to an instrument using predefined parameters without any user interaction after configuration starts.	TR_70, TR_74, TR_86, TR_104, TR_112
BR_38	XMANAI shall reduce the amount of interaction with the software so that users spend more time measuring and less time browsing through the software.	TR_71, TR_72, TR_86, TR_100, TR_104, TR_107, TR_112, TR_113, TR_120, TR_122, TR_123
BR_39	XMANAI should collect and standardize machining data.	TR_1, TR_3, TR_10, TR_11, TR_16, TR_12, TR_13, TR_93, TR_118,
BR_40	XMANAI shall be able to provide information to detect easily the sources of problems.	TR_75, TR_89, TR_90, TR_97, TR_101, TR_102, TR_106, TR_110, TR_111
BR_41	XMANAI shall create pop-up messages for user instructions, instrument alignment, profile change, etc.	TR_76, TR_97, TR_101, TR_106
BR_42	XMANAI shall keep historical records of the machine on-site.	TR_45, TR_102, TR_103
BR_43	XMANAI shall be able to analyze and communicate results.	TR_63, TR_69, TR_80, TR_91, TR_92, TR_95, TR_126, TR_127, TR_130
BR_44	XMANAI shall create predefined measurement routines and explain why they are the correct ones.	TR_5, TR_6, TR_7, TR_8, TR_9, TR_94
BR_45	XMANAI shall be able to reduce the number of iterations required with the computer by the user, so the data collection will be more efficient.	TR_2, TR_16, TR_54, TR_64
BR_46	XMANAI shall perform improved visualizations of item annotations and value logic.	TR_83, TR_84, TR_88, TR_89, TR_90

3.4 Technical Requirements across the User Journeys

The workflows that are to be followed at high-level by a Data Scientist, a Data Engineer and a Business User as the targeted stakeholders of XMANAI, have been previously presented as User Journeys in Section 2. The journey of each stakeholder is split into phases that contain different steps, organized as the sequence of all the possible events that a stakeholder goes through. In this section, the extracted user stories are mapped and assigned to a related phase of activity of the User Journeys using the horizontal User Story Mapping technique.

User Story Mapping is an agile software development approach that generally helps to present user stories into the context of the overall functionality of a system. The User Story Mapping process arranges the User Stories in two dimensions:

- Horizontal: representing the user journey (steps a user takes to perform actions in the system) and grouping User Stories into higher levels of functionality/activity.
- Vertical: demonstrating the priority of the User Stories for different system releases (this mapping is achieved through the MVP activities that are presented in Section 5).

Table 3-3 shows the alignment of technical requirements across the Data Scientist User Journey.

Table 3-3: Technical requirements alignment across Data Scientist User Journey

User Journey Step	Technical Requirement No.
Understand the data and the problem at hand	TR_1, TR_2, TR_3, TR_5, TR_6, TR_7, TR_8, TR_9, TR_10, TR_11, TR_12, TR_13, TR_14, TR_15, TR_16, TR_17, TR_18, TR_19, TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28, TR_29, TR_30, TR_31, TR_32, TR_33, TR_35, TR_36, TR_37, TR_38, TR_39, TR_40, TR_42, TR_43, TR_44, TR_50, TR_51, TR_52, TR_53, TR_54, TR_107
Prepare the data and handle problematic data cases	TR_41, TR_55, TR_56, TR_57, TR_58, TR_59, TR_60, TR_61, TR_62, TR_131, TR_132, TR_134
Collaboratively design AI pipelines	TR_65, TR_66, TR_67, TR_68, TR_69, TR_70, TR_71, TR_72, TR_73, TR_74, TR_75, TR_76, TR_77, TR_78, TR_79, TR_80, TR_81, TR_83, TR_86, TR_96, TR_101, TR_102, TR_114, TR_115, TR_116, TR_117, TR_133
Explain AI models/ results	TR_82, TR_85, TR_87, TR_88, TR_92, TR_93, TR_103, TR_105, TR_108, TR_109, TR_110, TR_111, TR_113, TR_118, TR_119, TR_120, TR_121, TR_122, TR_123
Evaluate AI models/ pipelines	TR_97, TR_98, TR_99, TR_100, TR_124, TR_125, TR_126, TR_128, TR_129

Table 3-4 shows the alignment of technical requirements across the Data Engineer User Journey.

Table 3-4: Technical requirements alignment across Data Engineer User Journey

User Journey Step	Technical Requirement No.
Collaborate on uploading data/Handling data	TR_1, TR_2, TR_3, TR_4, TR_5, TR_6, TR_7, TR_8, TR_9, TR_10, TR_11, TR_12, TR_13, TR_14, TR_15, TR_16, TR_19, TR_20, TR_28, TR_29, TR_30, TR_32, TR_34, TR_35, TR_36, TR_37, TR_38, TR_39, TR_40, TR_42, TR_44, TR_107
Collaborate on the design of AI pipelines	TR_65, TR_66, TR_67, TR_69, TR_70, TR_74, TR_96, TR_101, TR_102, TR_133
Deploy AI pipelines to production	TR_71, TR_72, TR_73, TR_82, TR_103, TR_104, TR_106, TR_108, TR_112

Table 3-5 shows the alignment of technical requirements across the Business User User Journey.

Table 3-5: Technical requirements alignment across Business User User Journey

User Journey Step	Technical Requirement No.
Provide data	TR_1, TR_2, TR_3, TR_12, TR_13, TR_14, TR_15, TR_16, TR_20, TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28, TR_29, TR_30, TR_31, TR_32, TR_33, TR_34, TR_36, TR_37, TR_38, TR_39, TR_40, TR_45, TR_46, TR_47, TR_48, TR_49
Contribute to better understanding and evaluating data/ AI models / Results	TR_35, TR_63, TR_64, TR_127
Gain insights into AI results	TR_84, TR_89, TR_90, TR_91, TR_94, TR_95, TR_105, TR_130

4 Early Data Perspectives and Requirements

In this section, early insights on the demonstrator data and open data are provided.

4.1 Overview

In the early phases of the project, the data acquisition activities have followed two approaches: (a) data acquisition from the XMANAI Demonstrators and (b) desk-based research for manufacturing open data sources.

To this end, Section 4.2 describes the relevant data sources from the XMANAI demonstrators, including Ford, Whirpool, CNH and Unimetrik, that will interface with XMANAI platform. This information is taken from the Trial Handbook Chapter 3 provided in WP6 taking into consideration that the present deliverable is public (thus removing any confidential information). Chapter 3 of Trial Handbook provides details and information about the profile of data sources that come from the involved, in place, legacy systems including the following information:

- *Accessibility*: Provides information on how data is accessible which can be through an API Endpoint, File Extract, SQL data access, OPC-UA SDK, etc.
- *Profile*: Descriptions the profile of data including data description, format (JSON, XML, etc.), volume, velocity (e.g. Batch of data, Real-time, Available upon request, etc.) and data encryption, etc.
- *Structure*: Details about structure of data sources provided by demonstrators (.e.g. the columns of data table in the case of structured data or syntax and semantics of data in the case of non-structured data).
- *Historical data*: Details about availability of historical data (e.g. time period covered, concerned factories).
- *Data Generation*: Provides information on how data is generated, whether generated in the real system, simulated, etc.

As the result of the desk-based research, a list of open data sources in the area of manufacturing is collected with the cooperation of all XMANAI partners. Section 4.3 of this deliverable provides details about the profile of the open access datasets that have been found.

4.2 Data Acquisition from the XMANAI Demonstrators

4.2.1 Data Acquisition for Demonstrator I – FORD

The data sources of the Ford demonstrator that will interact with the XMANAI platform are described in the following paragraphs. The information can be clustered in the following groups:

1. Data source #1: FIS
2. Data source #2: QLS-CM
3. Data source #3: Datamart

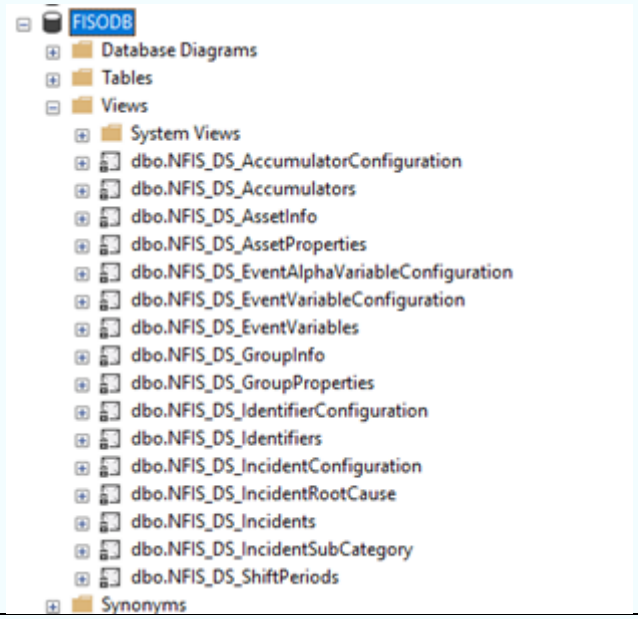
Data source #1: FIS

FIS is a computer-based data collection and reporting system that monitors Machine Performance Data, such as uptime, downtime, blocked and starved conditions, other machine states, machine faults and warnings, and other line conditions. FORD needs FIS in order to have data on machine status, and show the representation the current state. FORD also need this data to calculate some predictions in the next shifts. These data will be the input of the both FORD use cases described in D6.1.

The information is stored in a SQL database, and a priori we will have access to a copy of this DB with a few minutes delay.

The first trial is to create JSON files from the SQL database and send them via MQTT or an API to the XMANAI platform, other option is to create a gateway that reads the SQL database and sends the data to XMANAI platform.

Table 4-1: FORD Data Source #1 Profiling

<table>
<tr><td colspan="3">Data Involved name</td><td>Factory Information System (FIS)</td></tr>
<tr><td colspan="3">Type</td><td>Product data, machine data</td></tr>
<tr><td rowspan="9">Details</td><td colspan="2">Accessibility</td><td>The data is stored in a SQL database. For the time being we will have access to the data with a few minutes delay. A proxy account will need to be requested.</td></tr>
<tr><td rowspan="7">Data Profile</td><td>Description</td><td>• Machine status
• Faults and Warnings description
• Cycle counters and cycle times</td></tr>
<tr><td>Format</td><td>SQL database > JSON > to send XMANAI
SQL database > to send XMANAI</td></tr>
<tr><td>Volume</td><td>In assembly area near to 300 Assets (operations reporting to FIS).</td></tr>
<tr><td>Velocity</td><td>One part of the first use of case (representation) needs data in real-time or near to real time. For this reason, this data should be updated as quickly as possible.
It should be noted that the current data source does not store the data in real time and this will be available with a delay of a few minutes. Even so, the refresh rate should be around 20 seconds as this is the standard cycle time of the line.</td></tr>
<tr><td>Veracity</td><td>N.A.</td></tr>
<tr><td>Validity</td><td>Not required as it is a corporate system.</td></tr>
<tr><td>Volatility</td><td>Currently 1 year of data is stored in the corporate database.</td></tr>
<tr><td>Encryption</td><td>The data is not encrypted but we will require HTTPS to be sent to the platform.</td></tr>
<tr><td colspan="2">Historical Data</td><td>Save data of the last year, and only have the data of our plant (VEP)</td></tr>
<tr><td colspan="3">Data Structure</td><td>Available views
 
</td></tr>
<tr><td colspan="3">Sample of data</td><td><u>EXAMPLES OF QUERIES:</u>
/* NFIS_DS_AccumulatorConfiguration */
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category] ,[AccumulatorName] ,[Active] ,[AREA_SAKEY] ,[NODEID] ,[ACCUMULATORID] ,[ACCDEFINITIONID] ,[DATATYPEID]
FROM [FISODB].[dbo].[NFIS_DS_AccumulatorConfiguration]</td></tr>
</table>

```
 /* NFIS_DS_Accumulators */
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category]
,[AccumulatorName] ,[InsertTime] ,[Value] ,[AREA_SAKEY] ,[NODEID]
,[ACCUMULATORID] ,[ACCDEFINITIONID] ,[DATATYPEID]
FROM [FISODB].[dbo].[NFIS_DS_Accumulators]
Where InsertTime >= '2021-07-26 00:00:00'

 /* NFIS_DS_AssetInfo */
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[GroupName] ,[Order]
,[OrderTag] ,[BottleneckOrder] ,[Active] ,[AREA_SAKEY] ,[NODEID]
,[PARENTGROUPID]
FROM [FISODB].[dbo].[NFIS_DS_AssetInfo]

 /*NFIS_DS_AssetProperties */
SELECT [AREA_SAKEY] ,[NODEID] ,[Alias] ,[AreaPaypoint] ,[Asset
Type] ,[Asset_ID] ,[Bottleneck_order] ,[BottleneckPoint]
,[CellGroup1] ,[CellGroup2] ,[CellGroup3] ,[DERR] ,[DsgnCycleTime]
,[DsgnEquipCycleTime] ,[DsgnIndexCycleTime] ,[ERC] ,[GRR] ,[JPH]
,[KeyBuffer] ,[Labor Asset] ,[LaborAsset] ,[LERR] ,[LinePaypoint]
,[Node ID] ,[OEM] ,[OEMName] ,[Operator 1 Time] ,[Operator 2 Time]
,[Pallet Number] ,[Parallel Process] ,[PartsPerCycle] ,[PCON ID]
,[Production Critical Points] ,[ProductionSpeed] ,[Robot 1 Time]
,[Robot 2 Time] ,[Robot 3 Time] ,[Robot 4 Time] ,[Robot 5 Time]
,[Robot 6 Time] ,[Robot Type] ,[RunAtRateJPH] ,[SimulationMTBF]
,[SimulationMTTR] ,[Standard Cell Type] ,[StdCycleTime]
FROM [FISODB].[dbo].[NFIS_DS_AssetProperties]

 /* NFIS_DS_EventAlphaVariableConfiguration > Empty  */

/*NFIS_DS_EventVariableConfiguration*/
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[EventName]
,[VariableName] ,[Active] ,[AREA_SAKEY] ,[NODEID]
,[EVENTVARIABLEID] ,[EVENTID] ,[VARIABLEID]
FROM [FISODB].[dbo].[NFIS_DS_EventVariableConfiguration]

/* NFIS_DS_EventVariables */
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[EventName]
,[VariableName] ,[RecordTime] ,[Value] ,[AREA_SAKEY] ,[NODEID]
,[EVENTVARIABLEID] ,[EVENTID] ,[VARIABLEID]
FROM [FISODB].[dbo].[NFIS_DS_EventVariables]
Where RecordTime >= '2021-07-26 00:00:00'
/**NFIS_DS_GroupInfo**/
SELECT [Area] ,[GroupName] ,[Path] ,[Order] ,[OrderTag]
,[BottleneckOrder] ,[Active] ,[AREA_SAKEY] ,[GROUPID]
,[PARENTGROUPID]
FROM [FISODB].[dbo].[NFIS_DS_GroupInfo]

 /***NFIS_DS_GroupProperties**/
SELECT [AREA_SAKEY] ,[GROUPID] ,[Asynchronous Line]
,[Bottleneck_order] ,[GroupLinePoint] ,[GroupPayPoint]
,[GroupPayPointID]
FROM [FISODB].[dbo].[NFIS_DS_GroupProperties]

 /****** NFIS_DS_IdentifierConfiguration ******/
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category]
,[IdentifierName] ,[Active] ,[AREA_SAKEY] ,[NODEID] ,[IDENTIFIERID]
,[IDENTIFIERDEFINITIONID] ,[DATATYPEID]
FROM [FISODB].[dbo].[NFIS_DS_IdentifierConfiguration]

 /****** NFIS_DS_Identifiers  ******/
SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category]
,[IDENTIFIERNAME] ,[STARTTIME] ,[ENDTIME] ,[VALUE]
,[PropertyValue] ,[AREA_SAKEY] ,[NODEID] ,[IDENTIFIERID]
```

		FROM [FISODB].[dbo].[NFIS_DS_Identifiers] where ENDTIME >= '2021-07-26 00:00:00' **/****** NFIS_DS_IncidentConfiguration ******/** SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category] ,[IncidentDesc] ,[Active] ,[AREA_SAKEY] ,[NODEID] ,[INCIDENTID] ,[INCIDENTDEFINITIONID] ,[DATATYPEID] FROM [FISODB].[dbo].[NFIS_DS_IncidentConfiguration] **/****** NFIS_DS_IncidentRootCause > NOT USED ******/** SELECT [Area_SAKEY] ,[INCIDENTID] ,[STARTTIME] ,[ENDTIME] ,[Category] ,[REASONID1] ,[RootcauseLevel1] ,[REASONID2] ,[RootcauseLevel2] ,[REASONID3] ,[RootcauseLevel3] ,[REASONID4] ,[RootcauseLevel4] ,[REASONDESC] FROM [FISODB].[dbo].[NFIS_DS_IncidentRootCause] where ENDTIME >= '2021-07-26 00:00:00' **/****** NFIS_DS_Incidents ******/** SELECT [Area] ,[AssetName] ,[Path] ,[Alias] ,[Category] ,[Description] ,[StartTime] ,[EndTime] ,[TriggerValue] ,[Duration] ,[AREA_SAKEY] ,[NODEID] ,[INCIDENTID] ,[INCIDENTDEFINITIONID] ,[DATATYPEID] FROM [FISODB].[dbo].[NFIS_DS_Incidents] where ENDTIME >= '2021-07-26 00:00:00' **/**** NFIS_DS_IncidentSubCategory > Empty *****/** **/****** NFIS_DS_ShiftPeriods ******/** SELECT [Area] ,[Path] ,[ShiftName] ,[ShiftDescription] ,[PeriodName] ,[StartTime] ,[EndTime] ,[Duration] ,[ShiftDate] ,[ShiftStartTime] ,[ShiftEndTime] ,[IsProductive] ,[SummarizedCode] ,[AREA_SAKEY] ,[OBJECTID] ,[SHIFTID] ,[SHIFTPERIODID] FROM [FISODB].[dbo].[NFIS_DS_ShiftPeriods]
Dataset generation	Was the data monitored in a system with real users?	Yes
	If no, how the data has been generated?	-

Data source #2: QLS-CM

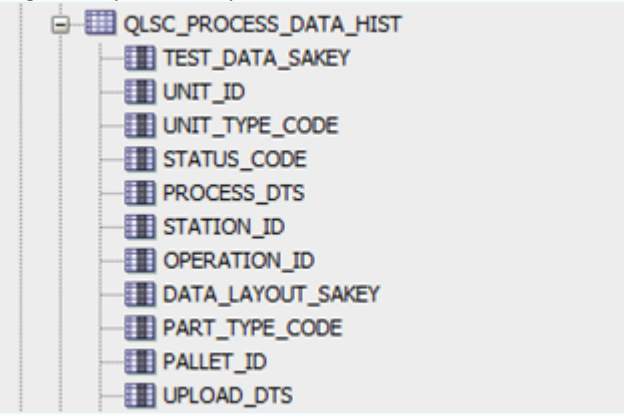
The Quality and Traceability system is used throughout PTO to collect birth history data for serialized assemblies and components. The purpose of this data source is to determe for the different components their build status, machining or assembly path, test status and quality status. It is a system that can alert operators of any quality problem.

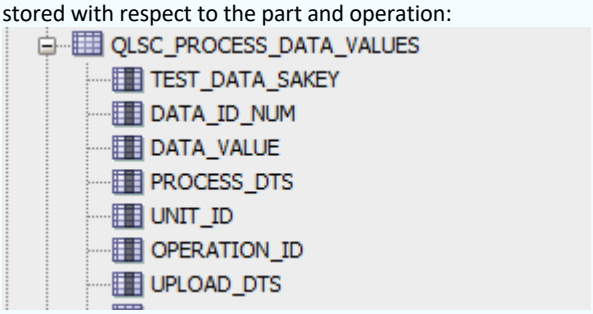
This data contains the serial number of the engine or component (Crankshaft, Camshaft, Cylinder Block and Cylinder Head) in the current operation for the traceability of the parts and also the quality data.

The information is stored in a SQL database and the data shall be sent to the XMANAI Platform via JSON files or MQTT.

Table 4-2: FORD Data Source #2 Profiling

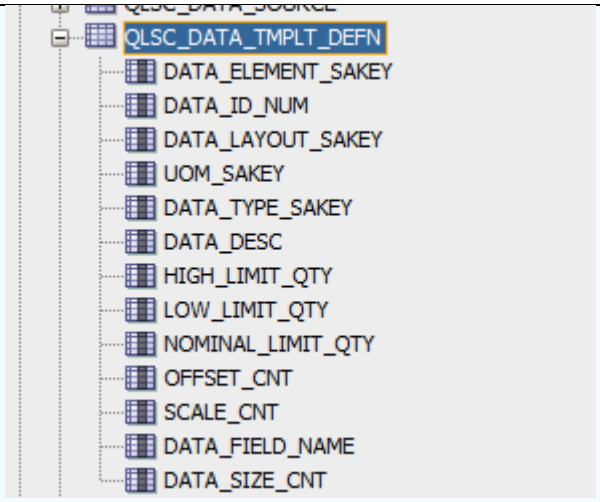
Data Involved name		**Quality Leadership System - Component Manufacturing (QLS-CM)**
Type		**Product data, process data, production logistics data**
Details	**Accessibility**	SQL Server

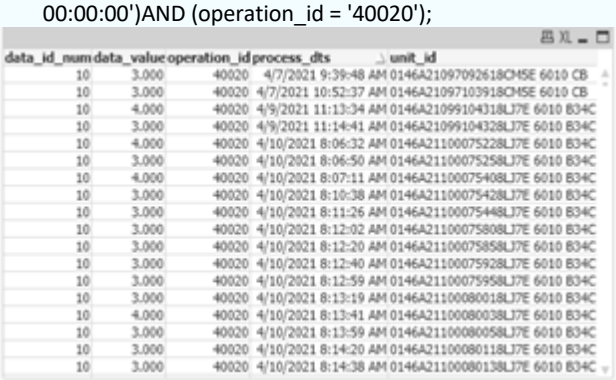
<table>
<tr><td rowspan="10"></td><td colspan="2"></td><td>JSON files
MQTT</td></tr>
<tr><td rowspan="8">Data Profile</td><td>Description</td><td>• Traceability of serial numbers
• Quality data</td></tr>
<tr><td>Format</td><td>SQL server > JSON</td></tr>
<tr><td>Volume</td><td>Queries by upload point</td></tr>
<tr><td>Velocity</td><td>Update near cycle time, because is data of parts</td></tr>
<tr><td>Veracity</td><td>-</td></tr>
<tr><td>Validity</td><td>-</td></tr>
<tr><td>Volatility</td><td>-</td></tr>
<tr><td>Encryption</td><td>-</td></tr>
<tr><td colspan="2">Historical Data</td><td>1 year</td></tr>
<tr><td colspan="3">Data Structure</td><td>In this system there are three main tables that store the relevant information.
Traceability [qlsc_process_data_hist]: in this table it can be seen at what time and through which machine each engine and engine component has passed.
 

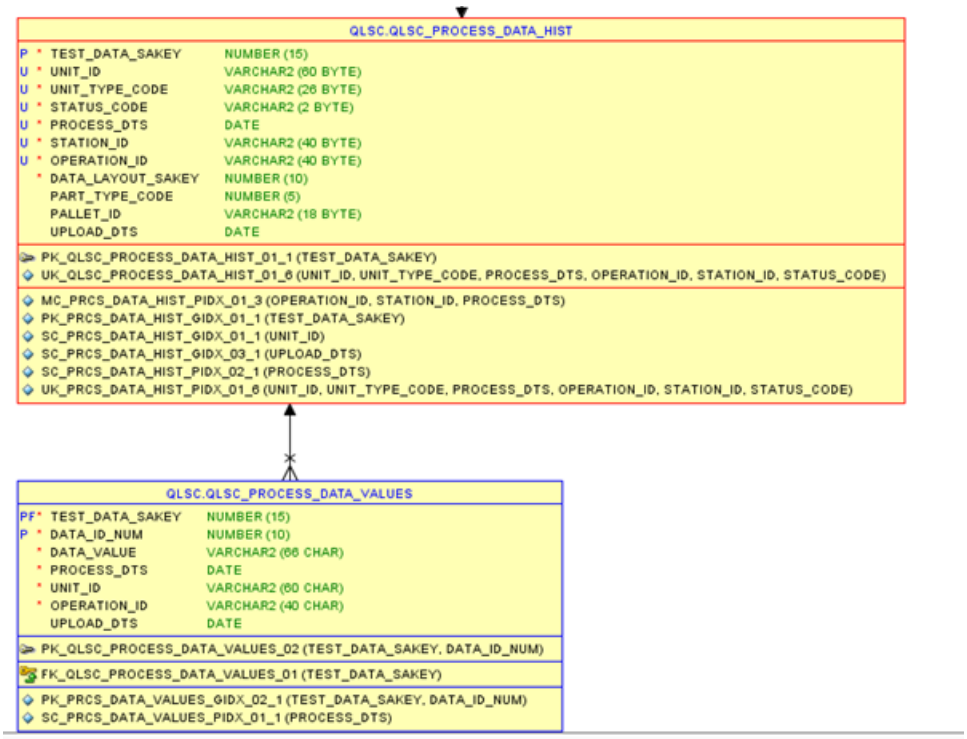
Meaning of main fields:
• unit_id: Serial Number
• unit_type_code: WERS (derivate)
• process_dts: Time stamp
• station_id/operation_id: Station and operation
Quality [qlsc_process_data_values] in this table quality data is stored with respect to the part and operation:
 

Meaning of main fields:
• data_id_num: field number
• data_value: quality data
• process_dts: Time stamp
• unit_id: Serial Number
• operation_id: operation

Mapping [qlsc_data_tmplt_defn]: Link the data values with the description of the data</td></tr>
</table>

<table>
<tr><td colspan="2"></td><td> 

Meaning of main fields:
• data_layout_sakey: link with the operation
• data_id_num: field number
• data_desc: description of the data</td></tr>
<tr><td colspan="2">Sample of data</td><td><u>EXAMPLES</u>
Data_values
SQL SELECT process_dts, unit_id , operation_id , upload_dts , plant, data_id_num, data_value FROM HIVE.dsc60082_qlscm_tz_db.qlsc_process_data_values WHERE plant = 'val' AND process_dts > ('2021-01-01 00:00:00')AND (operation_id = '40020');
 
</td></tr>
<tr><td rowspan="2">Dataset generation</td><td>Was the data monitored in a system with real users?</td><td>Yes</td></tr>
<tr><td>If no, how the data has been generated?</td><td>-</td></tr>
</table>



Data Source #3: Datamart

CMMS3 (Common Material Management System) is a mainframe computer system used to support Ford Assembly and Manufacturing Plants Worldwide. It is used to support the following operations: shipping, receiving, inventory, scheduling, releasing, bar coding, warehousing and accounting. This system is also used to support electronic communication between Ford, its suppliers and customers.

FORD use this system to view the demand, the production plan, the available parts and the components of different engines so as to plan the best mixt of production.

Table 4-3: FORD Data Source #3 Profiling

Data Involved name			Datamart
Type			Logistic data
Details	**Accessibility**		SQL-Server
	Data Profile	**Description**	• Production plan • Components of engine • Available parts
		Format	SQL database -> JSON
		Volume	-
		Velocity	Some queries per Day
		Veracity	-
		Validity	Not required as it is a corporate system.
		Volatility	-
		Encryption	The data is not encrypted but we will require HTTPS to be sent to the platform.
	Historical Data		Details for the available historical data (e.g. time period covered, concerned factories)
Data Structure			**EXAMPLES OF RELEVANT DATA AND DESCRIPTION:** dbp_CPNT023_part.csv --- Table with all parts information CD_PLANT.....Plant code NO_PART_PREFIX.....Part name prefix

	NO_PART_BASE.....Base of part name NO_PART_SUFFIX.....Part Name Suffix The part number is the unique identifier of a particular item. It consists of four fields: PREFIX (7 characters), BASE (9 characters), SUFFIX (8 characters) and an optional CONTROL CODE (3 characters). If the plant is a European Plant, the part number will be limited in size to size to: PREFIX (6 characters), BASE (8 characters), SUFFIX (8 characters), and CONTROL CODE (3 characters). optional CONTROL CODE (2 characters). If the plant is connected to the WIPS (Worldwide Integrated Purchasing Sys- tem) and the part number is a type of purchase part (PP, PM, RM, RM, PP, RM, RM), then the part number is a type of purchase part. (PP, PM, RM, CS or BK), the prefix, base and suffix must not exceed six (6), eight (8), eight (8), eight (8), eight (8), eight (8), and eight (8). six (6), eight (8), and eight (8) characters respectively. CD_PART_STATUS.....PART STATUS Data Element: CPNT023-PART CD-PART-STATUS Description: PART STATUS Part Status is an indicator of where a part is in its life cycle. These are valid Part Status Codes used: PRE-PRODUCTION U = Unapproved I = Warranty Initial Sample Approved N = New F = Functionality Approved L = Approved Colour P = Prototype STRUCTURED PARTS FILE A = Approved B = Cancellation C = Current Model R = Change for Out of Stock S = Service Only M = Change Required O = Obsolete (Incomplete Availability) D = End of Series (Complete Availability) CD_PART_TYPE..... part type Data Element: CPNT023-PART CD-PART-TYPE Description: PART TYPE The Part Type is a two-character code that identifies the part's characteristics of the part: AS - Assembly PM - Raw Material (metal) (control lift tag) CT - Packaging - Returnable/depository RM - Raw Material NC - Packaging - Non-returnable BK - Bulk CS - Consignment DU - Packing protection EI - End Item MO - Model/Option MP - Part To Be Manufactured PI - Purchase In (Process) Assembly

PP - Purchase Part

DS_PART.....Part Description.

Data Element: CPNT023-PART DS-PART
Description: PART DESCRIPTION

An alphanumeric text field explaining the part number. This field is maintained on the ADFA, Parts Maintenance screen as a 34-character field and appears only on other screens in CMMS3.
The Description may be abbreviated on some screens in CMMS3 if the field size for this data element is less than 34 characters.

QT_PRT_BOH_LOOSE.....Stock in plant

Data Element: CPNT023-PART QT-PRT-BOH LOOSE
Description: BOH (BALANCE AVAILABLE)
The loose BOH represents the total quantity of a plant part available to satisfy shipping and/or production requirements.
It is calculated by starting with the BOH "In Plant" and:
- Subtracting the Sufficient On-Line Quantity to deposit production parts (Assembly Only).
- Subtracting Rejects
+ Adding variations of manual pending cycles.

dbo_CSFT071_ACUMULATED.csv

Table with the daily production of each cost center.
NO_WORK_CNTR.....Cost centre
The physical location where parts are produced in a plant. This identification is unique to a department, area or plant.

DT_PERF_SCHD.....Transaction Date
QT_ORIG_SCHD.....Planned production quantity
QT_PROD.....Quantity Produced

The number of pieces reported as production for this Part/Work Centre/Department, for the specified date.

QT_SCRAP_FROM_PROD.....Parts scrap pulled
dbo_CSFT023_PART_WC.csv

Fields explained in previous tables.

Table with the relation of parts and their cost centers.
dbo_CSFT018_MSTR_SCHD.csv

Table with the future planning (4 weeks) of production by cost center.
Fields explained in previous tables.
DT_SCHD.....Facha of the plan.
QT_SCHDD.....Planned quantity.
CD_FLAG_MANUAL.....Plan creator.

The Scheduled Quantity is the number of parts to be produced on the indicated date.
The number has a suffix of an alphabetical character to highlight whether it has been created by system ("FLAG") or not,whether it has been created by system ("R"), whether it is a more frequently scheduled requirement ("M") or whether it has been created by system ("R").

dbo_CRCT004_CONVYNCE.csv

	----------CD_PLANT NO_CONVEYANCE Data Element: CRCT004-CONVYNCE NO-CONVEYANCE Description: VEHICLE NUMBER A number assigned by a carrier to this vehicle (for all modes of transport). CD_DUP_CONVEYANCE.....On cyclic routes route number within the cycle. QT_CONTAINERS.....Quantity of containers. NO_CONV_SEAL_1.....Container transport code. DT_SHIPPED.....The date when the shipment left the supplier's premises. DT_EXPECTED_ARRVL.....Date of arrival. The date on which a vehicle became in-transit for a plant. This field is automatically populated when the vehicle status is set to "I"(n transit) either from the ASN, or by manually creating a vehicle. If a vehicle is manually created, the In Transit date defaults to the current date. If the supplier does not send an expected arrival date in the ASN, the system generates a date using ADHA's transit history for the given mode of transport from the supplier's shipping point. If it is not available, the default values for the mode of transport will be used as follows: A - Air Transport 1 Day R - Rail 5 Days O - Ocean Freight 30 Days M - Road Freight 2 Days C - Consolidated Vehicle 4 Days CD_CONVEYANCE_STAT..... This field shows the current status of a vehicle in the plant. The status of the vehicle can be: "I"(n-transit - in transit), "B"(ullpen - temporary storage),"V"(erified - Verified), "C"(all-In - Request), or "A"(rrived - Arrived) CD_MODE_TRANSPRTN VALID Transport Modes: A = Air Freight B = Between Common Warehouses C = Common Truck/Grouping D = Group Departure/ODC/Consolidated E = Truck Dispatched (Pool/ODC/Consolidator to Plant) G = Rail/Highway L = Less than Truck M = Direct Truck O = Ocean Freight P = Consolidator (Transport to consolidator) R = Rail X = Charter S = Group Arrival/ODC/Cons (Vehicle Arrival to Pool/ODC/Consolidador) Z = "Road Railer" CD_CARRIER.....Carrier code NA_CARRIER.....Name of Carrier IN_CNVY_REPORTED DS_COMMENTS NO_BILL_OF_LADING.....

		Data Element: CRCT004-CONVYNCE NO-BILL-OF-LADING Description: SHIPPING AWARENESS NUMBER A document is used by a shipping line to acknowledge receipt of a freight and also serves as a contract for the movement of material.
Sample of data		Some examples are available as .csv files
Dataset generation	Was the data monitored in a system with real users?	Yes
	If no, how the data has been generated?	-

4.2.2 Data Acquisition for Demonstrator II – WHR

The data sources of the Whirlpool demonstrator that will interact with the XMANAI platform will be described below. The information can be clustered in the following groups:

- Transactional data
  - Historical Sales
- Behavioural data:
  - Google Analytics (clickstream and website events)
- Master Data:
  - Customers
  - Products & Product Hierarchy

Transactional Data

Table 4-4: WHR Data Source #1 Profiling

Data Involved name			HISTORICAL SALES
Type			Transactional Data
Details	**Accessibility**		Datasets are stored on Google Cloud Platform. They can be either directly downloaded from GCP GUI, or through Rest-API Endpoint provided by Google or through the "gsutil" command from the command line of Google Cloud SDK. For accessibility, a suitable Google project role is needed.
	Data Profile	**Description**	Standard SAP Sales Order, generated when a Product is purchased on our e-commerce platform. Sales orders related to the D2C business channels (Italy/Germany) and to the overall B2B traditional market (Italy/Germany)
		Format	Datasets are available as Google BigQuery tables.
		Volume	Volume of data is around 100 Megabyte at the moment.
		Velocity	Data related to sales are extracted on a daily basis. Additional info can be extracted on a weekly and/or monthly basis.
		Veracity	No bias for historical data.
		Validity	Datasets are correct and accurate for the intended use.
		Volatility	Tables collecting historical data do not expire. The time window will depend on the AI method employed for analysis: a time-window of around 30 months is generally required.
		Encryption	Datasets are encrypted at rest
	Historical Data		Data collected from 2021-04-01 onwards for D2C Italy, while from 2020-11-01 for D2C Germany. Currently, datasets show demand (and sales) data of around 1050 different products produced in 60 different manufacturing sources (WHR factories and OEM)
Data Structure			1. Market ID 2. Channel ID

		3. Order Date 4. Sales order ID 5. Customer ID 6. Product ID 7. Quantity 8. Price 9. Requested Delivery date 10. ....
Sample of data		-
Dataset generation	Was the data monitored in a system with real users?	Yes - data is generated by the SAP ERP system in the order-to-delivery management module.
	If no, how the data has been generated?	-

Behavioural data

Table 4-5: WHR Data Source #2 Profiling

Data Involved name			Google Analytics
Type			Clickstream Data
Details	**Accessibility**		Datasets are stored on Google Cloud Platform. They can be either directly downloaded from GCP GUI, or through Rest-API Endpoint provided by Google or through the "gsutil" command from the command line of Google Cloud SDK. For accessibility, a suitable Google project role is needed.
	Data Profile	**Description**	Clickstream data generated by Google Analytics tags embedded in our e-commerce platforms
		Format	Datasets are available as Google BigQuery tables. The format is not purely tabular, data is saved as nested array.
		Volume	>1TB at the moment, and growing
		Velocity	batch on daily basis
		Veracity	data is automatically generated and certified by Google analytics engine
		Validity	No bias expected
		Volatility	Data will not expire
		Encryption	Datasets are encrypted at REST
	Historical Data		Data availability since 01.04.2021 for D2C Italy, since 01.11.2020 for D2C Germany
Data Structure			1. Property ID 2. Visitor ID 3. Session ID 4. Visit Start Time 5. Date 6. Traffic Source (nested record) 7. Device (nested record) 8. Dimensions (nested record) 9. Hits (nested record)
Sample of data			...
Dataset generation		Was the data monitored in a system with real users?	No - No structured monitoring is in place by users
		If no, how the data has been generated?	The data is automatically generated by Google analytics on daily base according to the access to the commercial web site

Master Data

Table 4-6: WHR Data Source #3 Profiling

Data Involved name			CONSUMER AND PRODUCT MASTER DATA
Type			Master data
Details	**Accessibility**		Datasets are stored on Google Cloud Platform. They can be either directly downloaded from GCP GUI, or through Rest-API Endpoint provided by Google or through the "gsutil" command from the command line of Google Cloud SDK. For accessibility, a suitable Google project role is needed.
	Data Profile	**Description**	Traditional customer and product master data.
		Format	Datasets are available as Google BigQuery tables.
		Volume	Volume of data is around 2 Gigabytes at the moment.
		Velocity	Master Data related to product attributes and hierarchy are extracted on a weekly basis. Additional info can be extracted on a monthly basis.
		Veracity	Datasets can occasionally contain some mistakes on product codes and misclassification issues. Mistakes are monitored and corrected for the following extractions.
		Validity	Datasets are correct and accurate for the intended use.
		Volatility	Tables collecting master data do not expire.
		Encryption	Datasets are encrypted at REST
	Historical Data		Data collected from 2010
Data Structure			1. Customers • Market ID • Channel ID • Customer ID • Customer attributes 2. Product • Product ID • Product Description • Ean Code • Source • GPH (Global Product Hierarchy) • Product Attributes
Sample of data			-
Dataset generation		Was the data monitored in a system with real users?	Product Master Data coming from PRIME system (Whirlpool custom) Consumer Master Data coming from Customer DB (Whirlpool custom)
		If no, how the data has been generated?	-

4.2.3 Data Acquisition for Demonstrator III – CNH

The data sources of the CNH demonstrator that will interface with XMANAI platform are machineries which are present in the Modena plant shopfloor in an island consisting of 6 different machineries. Machineries are managed by shopfloor people in CNH. Some of them are in the shopfloor for 10 years while others are couple of years old. Majority of machineries are conned to the network while others will be connected during next months. The details of this data source are shown in the following table.

Table 4-7: CNH Data Source #1 Profiling

Data Involved name			Machinery data
Type			machine data
Details	**Accessibility**		File Extract/data flow
	Data Profile	**Description**	• 3 axes (X, Y, Z) frequency acceleration • 3 axes (X, Y, Z) frequency speed • Electrical panel • Air consumption • Pump • Mandrel

		Format	.xlsx/JSON
		Volume	A first dataset is available from 01/01/21 to 01/08/21; is it possible to access historical data from 2017
		Velocity	Batch, based on machine cycle, as the data are collected by sensors during execution of an empty cycle at the end of each production cycle
		Veracity	The comparability and veracity of data is assured by the fact that data are collected by sensors during execution of an empty cycle at the end of each production cycle. In this way sensors data are not affected by the differences of materials and work orders
		Validity	The data from sensors is correct and accurate for the intended use. Sometimes (couple of times per year) the sensors produce wrong values. This happens only for a single cycle and is correct the next cycle. Using not single values to trigger alarms but only averages of more than one the problem is solved.
		Volatility	Machinery is a stable configuration object so the data are valid for the whole life of the machinery and similar ones.
		Encryption	Inside CNH network data is not encrypted. Outside they are encrypted.
	Historical Data		Historical data for some machineries start from 2017, while for some machines is not present.
Data Structure			Kay-values doubles
Sample of data			Confidential Information
Dataset generation		Was the data monitored in a system with real users?	Yes (the user should approve the start of the empty measurement cycle)
		If no, how the data has been generated?	n/a

4.2.4 Data Acquisition for Demonstrator IV – UNIMETRIK

The data source of the UNIMETRIK demonstrator that will interact with the XMANAI platform is described in the following table.

Table 4-8: UNIMETRIK Data Source #1 Profiling

Data Involved name			Sensor data and 3D Metrological data
Type			Data Lake
Details	**Accessibility**		API REST
	Data Profile	**Description**	• Point clouds • GD&T • Color mapping • Machine conditioning • Model based design
		Format	TXT., XML (QIF), CSV, STEP
		Volume	4TB
		Velocity	150GB/day
		Veracity	High veracity
		Validity	Yes
		Volatility	-
		Encryption	-
	Historical Data		Past measurement projects
Data Structure			Heterogeneous datasets (Environmental parameters, machine configuration, operational data, CAD model, 3D scanner, CMM machine condition, process capacity)

Sample of data		-924.105408 -548.490723 -637.211487 0.001000 -1.000000 0.000000 -903.610962 -548.515625 -637.205994 -0.002000 -0.999998 -0.000000 -903.619324 -547.364929 -656.174072 -0.001000 -0.999999 0.001000 -923.386658 -547.304871 -656.172363 -0.010999 -0.999940 -0.000000 -916.044739 -547.947693 -650.394348 0.000000 0.001001 1.000000 -910.340149 -547.147278 -650.043152 -0.001001 -0.002002 0.999997 -904.204834 -547.154358 -645.035706 -0.999995 0.001000 0.003000 -904.658875 -547.144226 -643.253845 -0.999997 -0.002000 0.001000 -911.986450 -547.154358 -640.479980 -0.001000 -0.001000 -0.999999 -921.669983 -547.136108 -645.754456 0.999997 -0.001001 -0.002002 -3942.436768 -674.352173 -544.884705 -0.001000 -1.000000 -0.000000 -3929.637451 -674.434509 -544.885986 0.007000 -0.999976 0.000000 -3921.452637 -674.505005 -535.021057 -0.017997 -0.999838 -0.000000 -3950.178223 -674.343872 -531.702759 -0.000000 -1.000000 -0.000000 -3944.916504 -675.231445 -532.029419 1.000000 0.000000 0.000000 -3939.381348 -675.232483 -538.074402 0.019016 -0.002002 0.999817 -3932.939941 -675.241943 -538.649536 -0.009009 0.000000 0.999959 -3925.105957 -675.248047 -528.024414 -0.999995 -0.001000 0.003000 -3936.300293 581.411682 -551.565857 -0.106894 0.994268 -0.002187 -3938.322021 581.739685 -567.734985 0.014147 0.999900 -0.000180 -3960.459961 581.974670 -567.733704 -0.016880 0.999857 -0.000182 -3961.654785 582.994690 -556.335693 -0.355162 0.934805 -0.000191 -3950.550293 583.290771 -562.667114 0.001941 0.000181 0.999998 -3944.081787 582.684937 -561.063904 -0.001062 -0.001821 0.999998 -3939.599854 584.209839 -551.897705 -0.921405 -0.388603 0.000014 -3959.833008 582.382324 -551.911377 0.999999 -0.001135 0.000061 -896.427124 419.406006 -638.900391 -0.078935 0.996880 -0.000186 -919.893738 419.710266 -638.906555 -0.095903 0.995391 -0.000186 -906.957336 419.752747 -632.745728 0.001135 0.999999 -0.000181 -906.416077 419.136627 -651.036194 -0.155984 0.987760 -0.000189 -907.301941 420.403412 -648.185547 -0.001062 0.000181 0.999999 -911.047546 418.964661 -647.227356 0.003943 -0.001821 0.999991
Dataset generation	Was the data monitored in a system with real users?	Yes
	If no, how the data has been generated?	-

4.3 Data Acquisition from External Sources

This section focuses on the 22 open data sources found in the area of manufacturing and selected based on their relevance to the XMANAI demonstrators, to be eventually made available in the XMANAI platform.

In order to find the relevant open data sources, a desk-based search has been carried out with the cooperation of all XMANAI partners on open data sources. The following table provides details about the profile of open access datasets that are collected for this purpose.

Table 4-9: Open Manufacturing Data Sources Profiling

Title	Description	Type	Format	Domain	Licence	Provider	Statistics	URL
Machine failures	Sample dataset of one year hourly basis machine monitor, with the recorded info about failures.	TXT	CSV	Automotive Manufacturing	free	czuriaga	907.6 KB;28 fields;8784 instances	https://bigml.com/user/czuriaga/gallery/dataset/587d062d49c4a16936000810
Microsoft Azure Predictive Maintenance	Machine conditions and usage; Failure history; Maintenance history; Machine features	TXT	CSV	NA	free	Microsoft Azure	76.68 MB; 18 fields; 100-800k instances	https://www.kaggle.com/arnabbiswas1/microsoft-azure-predictive-maintenance
Versatile Production System	data taken of the Versatile Production System (VPS), which is part of the SmartFactory OWL. The VPS consists of several modules, starting with the delivery of material to be packaged (corn), storage of the material, dosing, filling, production (producing popcorn).	TXT	CSV	Condition Monitoring, Predictive Maintenance	CC BY-NC-SA 4.0	smart factory OWL	4.42 MB; 141 fiels; 11958 instances	https://www.kaggle.com/inIT-OWL/versatileproductionsystem?select=Filling_CapScrewer.module.csv
SECOM data set	Data from a semi-conductor manufacturing process including signals/variables collected from sensors and or process measurement points	TXT	CSV	Quality control in manufacturing	free	UC Irvine	5.4MB; 591 fields; 1567 instances	http://archive.ics.uci.edu/ml/datasets/SECOM
UK Manufacturers' Sales by Product Survey (PRODCOM)	Annual indicators on standard errors, response rates, revisions and any product code changes for the ProdCom survey, UK.	TXT	xls	Quality Manufacturing Product	Free	European Statistical Office (Eurostat)	1001.5 KB	https://www.ons.gov.uk/businessindustryandtrade/manufacturingandproductionindustry/datasets/ukmanufacturerssalesbyproductprodcomqualityindicators
FEMTO Bearing Dataset	Experiments on bearings' accelerated life tests	TXT	CSV		free	FEMTO-ST	1.1GB (zipped)	https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#femto
Bearing Dataset	Experiments on bearings	TXT	CSV		free	Center for Intelligent Maintenance Systems (IMS), University of Cincinnati	1 GB (zipped)	https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#bearing
Production line performance	Measurements of parts as they move through Bosch's production lines for quality control	TXT	CSV		free	Bosch	695 MB	https://www.kaggle.com/c/bosch-production-line-performance/data

Title	Description	Type	Format	Domain	Licence	Provider	Statistics	URL
Process control	Data from a multi-stage continuous flow manufacturing process over several hours timespan	TXT	CSV		free	Liveline Technologies	8 MB	https://www.kaggle.com/supergus/multistage-continuousflow-manufacturing-process
DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS	Supply Chain	CATEGORICAL/CONTINUOUS/BINARY	CSV	Provisioning , Production , Sales , Commercial Distribution	CCO: Public domain	Fabian Constante	182 MB; 63 fields; 180519 instances	https://www.kaggle.com/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis?select=DataCoSupplyChainDataset.csv
Quality Prediction in a Mining Process	Explore real industrial data and help manufacturing plants to be more efficient	CONTINUOUS/ DATE	CSV	Industry Manufacturing	CCO: Public domain	EduardoMagalhãesOliveira	175MB; 24 fields; 737453 instances	https://www.kaggle.com/edumagalhaes/quality-prediction-in-a-mining-process
Multi-stage continuous-flow manufacturing process	Real process data to predict factory output	CAREGORICAL/ CONTINUOUS/ DATE	CSV	Manufacturing process control	public	Liveline Technologies	8.11MB; 116 fields; 14088 instances	https://www.kaggle.com/supergus/multistage-continuousflow-manufacturing-process
Detecting Anomalies in Wafer Manufacturing	Detecting Anomalies using Machine Learning	CATEGORICAL/ CONTINUOUS	CSV	manufacturers of wafers(semiconductors)	public	ask9	5,27MB; 1559 fields;	https://www.kaggle.com/arbazkhan971/anomaly-detection
– Mercedes-Benz greener manufacturing	Reduce the time on the test bench.	CATEGORICAL/BINARY	CSV	Automotive Manufacturing	free	mercendes - benz	3.07 MB; 377 Fields; 4210 instances	https://www.kaggle.com/c/mercedes-benz-greener-manufacturing/data
Milling Data Set	Experiments on a milling machine		MAT		free	UC Berkeley	15MB	https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/#milling
Predictive maintenance in elevator industry.	Datasets from a variety of IoT sensors for predictive maintenance in elevator industry.	timeseries sampled at 4Hz	CSV	Manufacturing sensor data	Open	Huawei Munich Research Center.	2.5 MB	https://zenodo.org/record/3653909
Turbofan Engine Degradation Simulation Data Set	Engine degradation simulation was carried out using C-MAPSS. Four different were sets simulated under different combinations of operational conditions and fault modes. Records several sensor channels to characterize fault evolution.	CATEGORICAL	TXT	Industry Manufacturing	Open	NASA Ames	45,3 MB	https://ti.arc.nasa.gov/c/6/

Title	Description	Type	Format	Domain	Licence	Provider	Statistics	URL
Predictive Maintenance for Heating, Ventilation and A/C system	Predictive Maintenance Scheduling Optimization of Building Heating, Ventilation, and Air Conditioning Systems	CATEGORICAL	XLS	Industry Manufacturing	CC BY NC 3.0	Mendeley Data	0,5 MB	https://data.mendeley.com/datasets/468ynm7rfz/1
Steel Plates Faults Dataset	A dataset of steel plates' faults, classified into 7 different types.	CATEGORICAL	TXT	Industry Manufacturing	Open	University of California Irvine	1 MB	http://archive.ics.uci.edu/ml/datasets/Steel+Plates+Faults
Intel Lab data	data collected from 54 sensors deployed in the Intel Berkeley Research lab	CATEGORICAL	TXT	Industry Manufacturing	Open	Intel Labs	4 MB	https://www.kaggle.com/caesarlupum/iot-sensordata
Compressor on Aker BP's Valhall oil platform	Data from single compressor on Aker BP's Valhall oil platform in the North Sea.The data set available in the Cognite Data Platform includes time series data, maintenance history, and Process & Instrumentation Diagrams (P&IDs) for Valhall's first stage compressor and associated process equipment: first stage suction cooler, first stage suction scrubber, first stage compressor and first stage discharge coolers.In addition, data from the compressor's lubrication system, dry gas seal system and condition monitoring system (temperature and vibration) will be available.	time series	CSV	Sensor Data	Open	Open Industrial Data initiative (Cognite AS)	live streams	https://www.openindustrialdata.com/data/
Degration of a cutting blade	The Vega shrink-wrapper from OCME is deployed in large production lines in the food and beverage industry. The machine groups loose bottles or cans into set package sizes, wraps them in plastic film and then heat-shrinks the plastic film to combine them into a package. The plastic film is fed into the machine from large spools and is then cut to the length needed to wrap the film around a pack of goods. The cutting assembly is an important component of the machine to meet the high availability target.	Time-series	CSV	Manufacturing	CC BY-SA 3.0	inIT	109 MB / 518 files / 4671 cols	https://www.kaggle.com/inIT-OWL/one-year-industrial-component-degradation

Title	Description	Type	Format	Domain	Licence	Provider	Statistics	URL
	Therefore, the blade needs to be set-up and maintained properly.							
AI4I 2020 Predictive Maintenance Dataset Data Set	Since real predictive maintenance datasets are generally difficult to obtain and in particular difficult to publish, we present and provide a synthetic dataset that reflects real predictive maintenance encountered in industry to the best of our knowledge.	Multivariant; time-series	CSV	Manufacturing	open with citation (for publication)	HTW Berlin	509 KB / 1 file / 14 cols	https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset
BitcoinHeistRansomwareAddressDataset	We have downloaded and parsed the entire Bitcoin transaction graph from 2009 January to 2018 December. Using a time interval of 24 hours, we extracted daily transactions on the network and formed the Bitcoin graph. We filtered out the network edges that transfer less than B0.3, since ransom amounts are rarely below this threshold.	Multivariant; time-series	CSV	Cryptocurrency	open with citation (for publication)	University of Texas; University of Manitoba	113 MB / 1 file / 10 cols	https://archive.ics.uci.edu/ml/datasets/BitcoinHeistRansomwareAddressDataset#
Gas sensor array under dynamic gas mixtures Data Set	This data set contains the acquired time series from 16 chemical sensors exposed to gas mixtures at varying concentration levels. In particular, we generated two gas mixtures: Ethylene and Methane in air, and Ethylene and CO in air. Each measurement was constructed by the continuous acquisition of the 16-sensor array signals for a duration of about 12 hours without interruption.	Multivariant; time-series	structured text	Manufacturing	open with citation (for publication)	University of California San Diego	360 MB / 2 files / 19 cols	https://archive.ics.uci.edu/ml/datasets/Gas+sensor+array+under+dynamic+gas+mixtures
The broken machine	This dataset is about accumulated production machine data. There are about 60 different indicators and all of them unnamed. And there is of course data on machine breakdowns given in ytrain.csv file. Thus the topic of this dataset is to create a classification model to predict breakdowns. And, if there is need, to reveal the indicators with the greatest impact.	Multivariant	CSV	Manufacturing	CC0	Ivan Loginov	486 MB / 2 files / 59 cols	https://www.kaggle.com/ivanloginov/the-broken-machine

5 XMANAI Draft Minimum Viable Product (MVP)

This section describes the approach followed step-by-step in order to consolidate the XMANAI Minimum Viable Product (MVP) in its early release.

5.1 Overview

The Minimum Viable Product (MVP) refers to a version of a product with the minimum set of features and functionalities that can satisfy early adopters who, in turn, can promptly provide feedback for future product improvements. This concept, as expressed originally in 2001 by Frank Robinson, CEO of SyncDev Inc., helps lean product development by suggesting a sweet spot between Return of Investment and Risk of Failure, which correlates directly to effort and time to market. According to Ries (2009), the MVP strategy "allows a team to collect the maximum amount of validated learning about customers with the least effort".

In practice, MVP is an iterative process that swiftly moves towards the prototyping phase without investing effort on elements that could hamper the overall development due to low user acceptance, lack of alignment to the actual users' needs and high complexity. In each iteration, risky assumptions are identified and tested, and useful feedback is collected that steers product design and development to the proper direction.

For XMANAI, the MVP represents the overall mindset and strategy adopted for distributing efficiently the development and integration workload, for continuously testing the end user reaction, and validating the methodological ideas and hypothesis. As such, the XMANAI MVP will guide the research and development activities of the project throughout its lifecycle and will be maintained as a 'live' document which will continuously be updated with new research findings and feature requests with added value. The starting point for this journey is the definition of the draft XMANAI MVP which will be covered in detail in the subsections to follow. The goal is to identify and extract a compact set of features to be initially implemented, which will be prioritized based on the assessment of their added value from both the business and technical perspective. The following figure, Figure 5-1, presents in brief the XMANAI MVP definition approach, which involves three core phases, namely Feature Definition, Feature Assessment and MVP Consolidation and runs over the three iterations of WP1.

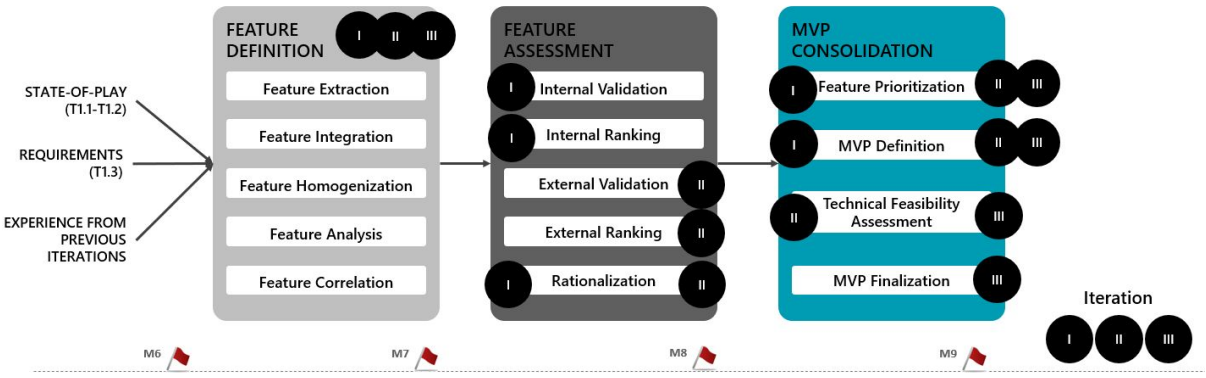


Figure 5-1: XMANAI MVP Approach

In this context, it needs to be noted that even if the MVP pinpoints the minimum set of features that are necessary for a product to be deployed and validated, it does not dictate the XMANAI consortium to seize their work when reaching that state; on the contrary, the MVP is a strategy, focusing on the most valuable assets, while taking into account complexity constraints that could push some features for later stages of the project in order to provide time for research or reach a satisfying level of maturity for the other features.

The MVP will be then validated and updated from the different activities in WP1, WP2-WP4, WP5 and WP6 through a set of interviews and questionnaires as part of a light "market-research" study, which will further streamline the initial release. Through the feedback acquired by the demonstrator partners, as well as the technical partners and external stakeholders, the consortium will consistently work towards finalizing the proposed features and functions, in order to improve the overall platform and deliver a final solution that facilitates to the highest possible degree the XMANAI concept and methodology.

5.2 Feature Elaboration

Based on the technical requirements presented in Section 3, a user story mapping exercise has been performed to extract MVP features (or epics) based on the detailed technical requirements backlog that was presented in Section 3.2. The resulted 45 features are presented below.

XMANAI_F_UM_001. Organization-based access

Description	In XMANAI platform, users within the same organization can access all data assets that belong to that organization.
Category	I. User Management
Related Requirements	N/A
Prerequisites	-

XMANAI_F_UM_002. Project-based access

Description	In XMANAI platform, users from different organizations can access data assets that are grouped under a project in order to ensure that users from different organizations can contribute for a common purpose/business problem.
Category	I. User Management
Related Requirements	N/A
Prerequisites	-

XMANAI_F_UM_003. Delete all data assets

Description	The XMANAI platform should allow for the deletion of all data assets that belong to an entity (organization or individual) as per GDPR guidelines regarding the 'right to be forgotten'.
Category	I. User Management
Related Requirements	N/A
Prerequisites	-

XMANAI_F_DI_004. Data Sources Management

Description	The XMANAI platform should allow for the addition, update, deletion, configuration and scheduling of various types of data sources (files, APIs, etc) out of which data will be ingested.

Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_1, TR_2, TR_4, TR_15, TR_54
Prerequisites	-

XMANAI_F_DI_005. Data Secure Uploading as file(s)

Description	The XMANAI platform should allow for the safe and reliable uploading of data as a single file or multiple files.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_3
Prerequisites	-

XMANAI_F_DI_006. Data Secure Uploading via API

Description	The XMANAI platform should allow for the safe and reliable uploading of data via an API.
Category	II. Data Ingestion (User Journey(s): Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_18
Prerequisites	-

XMANAI_F_DI_007. Data Mapping to a Data Model and Harmonization

Description	The XMANAI platform should provide common data models for all incoming datasets to adhere to in order to ensure explainability at data level. This mapping process will also perform harmonization techniques, such as data type casting, transformation to common measurement unit/timestamp.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_13, TR_54
Prerequisites	-

XMANAI_F_DI_008. Data Model Management

Description	The XMANAI platform should allow for data model management, in order to view the different models per domain/problem, add/edit/remove their contents, keep appropriate versions and extract them into different representations.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1)
Related Requirements	TR_5, TR_6, TR_7, TR_8, TR_9, TR_10, TR_11
Prerequisites	-

XMANAI_F_DI_009. Data Cleansing

Description	The XMANAI platform should provide a cleansing mechanism and allow for quality checks and the definition and execution of cleaning rules before storing the data.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_14
Prerequisites	-

XMANAI_F_DI_010. Data Anonymisation

Description	The XMANAI platform should provide an anonymization mechanism and allow for the definition and execution of anonymization rules before storing the data.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_29, TR_30
Prerequisites	-

XMANAI_F_DI_011. Data Storage in central XMANAI Cloud Storage

Description	The XMANAI platform should allow for the storage of data in centralized servers in the cloud.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_3, TR_31
Prerequisites	-

XMANAI_F_DI_012. Data Storage on-premise / in private cloud

Description	The XMANAI platform should allow for the storage of data in on-premise devices and/or in private cloud spaces per demonstrator.
Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	TR_16, TR_42
Prerequisites	-

XMANAI_F_DI_013. Data Encryption

Description	The XMANAI platform should allow for the encryption of data while a) transferred onto the platform and/or b) stored on the platform

Category	II. Data Ingestion (User Journey(s): Business User Phase 1, Data Scientist Phase 1, Data Engineer Phase 1)
Related Requirements	BR_24
Prerequisites	-

XMANAI_F_DM_014. Dataset Management

Description	The XMANAI platform should provide the means for complete management of data assets, like adding a dataset, editing or removing it. Appending metadata to better describe these assets and keeping versions of datasets are also part of this feature.
Category	III. Data Asset Management and Security (User Journey(s): Business User Phase 2, Data Scientist Phase 2, Data Engineer Phase 1)
Related Requirements	TR_12, TR_43, TR_64
Prerequisites	XMANAI_F_DI_011

XMANAI_F_DM_015. Features Management

Description	The XMANAI platform should provide functionalities for storing and managing curated features, allowing for the addition, editing or removal of features, appending metadata to better describe these assets and keeping versions of features.
Category	III. Data Asset Management and Security (User Journey(s): Data Scientist Phase 2, Data Engineer Phase 1)
Related Requirements	TR_44, TR_74
Prerequisites	-

XMANAI_F_DM_016. Results Management

Description	The XMANAI platform should allow for the editing or removal of results of different AI models and pipelines. Appending metadata to better describe these assets and keeping versions of them can also be part of this feature.
Category	III. Data Asset Management and Security (User Journey(s): Data Scientist Phase 2, Data Engineer Phase 1)
Related Requirements	TR_44
Prerequisites	-

XMANAI_F_DM_017. AI Model Management

Description	The XMANAI platform should provide for AI model management to create/store/update/delete/clone/configure/export/import/register AI models, and keep separate versions of each configuration and experiment associated with them.

Category	III. Data Asset Management and Security (User Journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_44, TR_118, TR_119, TR_120, TR_121
Prerequisites	-

XMANAI_F_DM_018. AI Pipeline Management

Description	The XMANAI platform should provide for AI Pipeline management to create/store/update/delete/clone/configure/export/import/join pipelines, to keep separate versions of each, and define templates for re-use. In addition, the platform should provide a view with the IPRs of all the assets involved.
Category	III. Data Asset Management and Security (User Journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_44, TR_49, TR_70, TR_71, TR_72
Prerequisites	-

XMANAI_F_DM_019. Data Asset Export

Description	The XMANAI platform should allow for exporting data assets (datasets, results) via file download or via API, depending on the IPR.
Category	III. Data Asset Management and Security (User Journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_17, TR_19
Prerequisites	XMANAI_F_DI_011

XMANAI_F_DM_020. Data Asset Access Policies and Licencing

Description	The XMANAI platform should allow users to manage and configure data asset access policies and licencing rules that are enforced at run-time across all its operations.
Category	III. Data Asset Management and Security (User Journey(s): Business User Phase 2, Data Scientist Phase 3)
Related Requirements	TR_21, TR_22, TR_23, TR_24, TR_25, TR_26, TR_27, TR_28
Prerequisites	-

XMANAI_F_DM_021. Data Asset Secure Transfer

Description	The XMANAI platform should provide the means to operate and enforce the secure transfer of any data asset through all platform layers (e.g. centralized cloud, private cloud, on-premise as they are to be detailed in the XMANAI Deliverable D5.1).
Category	III. Data Asset Management and Security (User Journey(s): Business User Phase 1, Data Scientist Phase 2, Data Scientist Phase 3)

Related Requirements	TR_31
Prerequisites	-

XMANAI_F_DM_022. Data Asset Access and Activity Logging

Description	The XMANAI platform should keep track of any activity related to data assets and monitor the access granted to different users.
Category	III. Data Asset Management and Security (User Journey(s): Business User Phase 1, Business User Phase 2, Data Engineer Phase 1, Data Scientist Phase 2, Data Scientist Phase 3)
Related Requirements	TR_45, TR_46, TR_47, TR_48
Prerequisites	XMANAI_F_DM_020

XMANAI_F_DM_023. AI Model Security Assessment

Description	The XMANAI platform should allow for model security assessment by testing the data sets, detecting and filtering out possible poisoned data points, and generating adversarial examples that can be added to the training set.
Category	III. Data Asset Management and Security (User journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_131, TR_132, TR_133
Prerequisites	-

XMANAI_F_DM_024. AI Pipeline Security Assessment

Description	The XMANAI platform should allow for security assessment of the overall AI pipeline, e.g. by testing the training data sets for possible unfair biases.
Category	III. Data Asset Management and Security (User journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_134
Prerequisites	-

XMANAI_F_DS_025. Data Asset Sharing

Description	The XMANAI platform should provide a contract-based sharing mechanism for data assets to grant legitimate access to selected users/organisations.
Category	IV. Data Asset Sharing/ Contracts (User journey(s): Business User Phase 2, Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_32
Prerequisites	-

XMANAI_F_DS_026. Data Asset Trading

Description	The XMANAI platform should provide a contract-based trading mechanism for data assets with selected users/organisations, and support offline, as well as online, payment methods.
Category	IV. Data Asset Sharing/ Contracts (User journey(s): Business User Phase 2, Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_33, TR_38
Prerequisites	-

XMANAI_F_DS_027. Data Asset Contract Management

Description	The XMANAI platform should provide functionalities related to contract management, that supports contract preparation, negotiation, approval and enforcement.
Category	IV. Data Asset Sharing/ Contracts (User journey(s): Business User Phase 2)
Related Requirements	TR_39, TR_40
Prerequisites	-

XMANAI_F_DS_028. Data Asset Search and Discovery

Description	The XMANAI platform should provide for data asset search and discovery, that includes indexing, sorting and filtering data assets, as well as viewing their details.
Category	IV. Data Asset Sharing/ Contracts (User journey(s): Business User Phase 2, Data Scientist Phase 2)
Related Requirements	TR_34, TR_35, TR_36, TR_41
Prerequisites	-

XMANAI_F_DS_029. Secure Transfer of data assets acquired to their legitimate consumers

Description	The XMANAI platform should provide the appropriate mechanism for the secure transfer of a data asset to a new legitimate data consumer that acquired it and keep the relevant history logs.
Category	IV. Data Asset Sharing/ Contracts (User journey(s): Business User Phase 2)
Related Requirements	TR_37, TR_38
Prerequisites	XMANAI_F_DM_020

XMANAI_F_DP_030. Data View & Visualisation

Description	The XMANAI platform should provide useful data views and visualisations, including data distribution, raw data sample preview, basic statistics, aggregations, descriptive analytics and exploratory queries over the data.

Category	V. Data Preparation (User journey(s): Data Scientist Phase 2, Business User Phase 2)
Related Requirements	TR_50, TR_51, TR_52, TR_53, TR_63
Prerequisites	XMANAI_F_DI_011

XMANAI_F_DP_031. Data Manipulation

Description	The XMANAI platform should allow for the most common data manipulation functionalities, such as merging, splitting, augmenting, resampling and aggregating data. The creation of new features (columns) and the handling of missing values are also considered to be part of the data manipulation functionalities.
Category	V. Data Preparation (User journey(s): Data Scientist Phase 2)
Related Requirements	TR_55, TR_56, TR_59, TR_62
Prerequisites	XMANAI_F_DI_011

XMANAI_F_DP_032. Data Transformation

Description	The XMANAI platform should allow for the most common data transformation functionalities, such as normalization, encoding, data type modification, etc.
Category	V. Data Preparation (User journey(s): Data Scientist Phase 2)
Related Requirements	TR_57, TR_58, TR_60, TR_61
Prerequisites	XMANAI_F_DI_011

XMANAI_F_AI_033. AI Model Design

Description	The XMANAI platform should allow users to design and configure an AI model based on a selected algorithm or by importing a model created with external AI/analytics tools/libraries.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 3)
Related Requirements	TR_94, TR_110, TR_124
Prerequisites	XMANAI_F_DM_017

XMANAI_F_AI_034. AI Model Training, Application & Evaluation

Description	The XMANAI platform should provide the appropriate mechanisms for AI model training, application and evaluation. This includes parameter configuration and optimisation, evalution metrics definition and monitoring, as well as check points saving. The differences between the experimentation stage and the production stage should be considered during the process.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 3)
Related Requirements	TR_75, TR_85, TR_95, TR_97, TR_98, TR_99, TR_100, TR_125, TR_126

Prerequisites	XMANAI_F_AI_033

XMANAI_F_AI_035. AI Pipeline Design

Description	The XMANAI platform should provide appropriate AI pipeline design functionalities, where a complete pipeline can be defined (as workflow) and configured. It should also allow for the addition of annotations / comments / explanations, the selection of appropriate packaging (for production) and the re-use of features in the form of templates that can be very common in various pipelines.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_65, TR_66, TR_67, TR_68, TR_74, TR_79, TR_80, TR_85, TR_87, TR_96, TR_114, TR_115, TR_116
Prerequisites	XMANAI_F_DM_018

XMANAI_F_AI_036. AI Pipeline Execution & Evaluation on XMANAI Common Cloud

Description	The XMANAI platform should provide the appropriate cloud execution environment for AI pipeline execution and evaluation. This includes the overall run and evaluation, retraining, scheduled runs, and automatic tests for AI models. There should be different setups for the experimentation stage and the production stage.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 5, Data Engineer Phase 3)
Related Requirements	TR_73, TR_86, TR_104, TR_105, TR_112
Prerequisites	XMANAI_F_DM_035

XMANAI_F_AI_037. AI Pipeline Execution & Evaluation on Premise / Private Cloud

Description	The execution and evaluation of an AI pipeline should be also available to run on premise or on a private cloud, either during experimentation or production. The configuration of this process involves scheduled runs, evaluation and retraining.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Business User Phase 2, Data Scientist Phase 5, Data Engineer Phase 3)
Related Requirements	TR_73, TR_75, TR_86, TR_104, TR_105, TR_108, TR_112, TR_113
Prerequisites	XMANAI_F_DM_035

XMANAI_F_AI_038. Collaboration over AI pipelines creation

Description	The XMANAI platform should provide a collaboration space for the comparison of AI pipelines experiments and models' performance. It will also maintain a history of the various events and provide a simulation environment for different settings, models and methods for the same task.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 3, Data Engineer Phase 2)

Related Requirements	TR_69, TR_103, TR_126, TR_128, TR_129
Prerequisites	XMANAI_F_DM_035

XMANAI_F_AI_039. AI Pipeline Results Management

Description	The XMANAI platform should allow for result management that views, stores,and exports AI pipeline results, providing summary statistics and comparisons with real values (for predictions). The result assets should be retrievable via API.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 4)
Related Requirements	TR_81, TR_82, TR_110, TR_111, TR_128, TR_129
Prerequisites	XMANAI_F_DM_037

XMANAI_F_AI_040. AI Pipeline Results Visualisation

Description	The XMANAI platform should provide visualization capabilities for the results with a list of various useful charts and graphs for the user to select. Adding comments/notes and saving or exporting the visualisations are also part of this feature.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Business User Phase 3, Data Scientist Phase 4)
Related Requirements	TR_83, TR_88, TR_89, TR_90, TR_113
Prerequisites	XMANAI_F_DM_037

XMANAI_F_AI_041. AI Pipeline Support

Description	The XMANAI platform should provide a set of supporting functionalities related to the AI pipeline execution, which include, but not limited to, recommendations and guidelines, execution logs, error handling, notifications and computational resources monitoring.
Category	VI. Data AI Analytics/Pipelines (User journey(s): Data Scientist Phase 3, Data Engineer Phase 2)
Related Requirements	TR_76, TR_93, TR_101, TR_102, TR_106, TR_107
Prerequisites	XMANAI_F_DM_035, XMANAI_F_DM_037

XMANAI_F_EX_042. Explainability Methods Management

Description	The XMANAI platform should effectively manage explainability methods, in terms of adding, removing, configuring, registering/importing different explainability techniques in AI pipelines and models.
Category	VII. AI Model/Results Explainability (User journey(s): Business User Phase 3, Data Scientist Phase 4)
Related Requirements	TR_77, TR_78, TR_117, TR_122, TR_123

Prerequisites	XMANAI_F_AI_033, XMANAI_F_AI_035

XMANAI_F_EX_043. Collaboration over AI explanations

Description	The XMANAI platform should allow for the cooperation of different stakeholders on the application of explainability methods at AI pipeline or AI model level. Querying the results, requesting for more details and adding notes/comments are key functionalities to ensure that explanations will be appropriate for the target stakeholders (business users).
Category	VII. AI Model/Results Explainability (User journey(s): Business User Phase 3, Data Scientist Phase 4)
Related Requirements	TR_91, TR_92
Prerequisites	XMANAI_F_AI_033, XMANAI_F_AI_035, XMANAI_F_AI_033, XMANAI_F_EX_042, XMANAI_F_EX_044

XMANAI_F_EX_044. Explainability Results Visualisation

Description	The XMANAI platform should provide comprehensible visualisations of the explainability results, such as charts and graphs, adjusted in accordance to the user profile (i.e. business user vs. data scientist)
Category	VII. AI Model/Results Explainability (User journey(s): Business User Phase 3, Data Scientist Phase 4)
Related Requirements	TR_84
Prerequisites	XMANAI_F_AI_033, XMANAI_F_AI_035, XMANAI_F_AI_033, XMANAI_F_EX_042

XMANAI_F_EX_045. Explainability Results Evaluation

Description	The XMANAI platform should allow for manual feedback and validation of the explainability results by the end users in order to improve the explanations provided per AI pipeline or model.
Category	VII. AI Model/Results Explainability (User journey(s): Business User Phase 3)
Related Requirements	TR_127
Prerequisites	XMANAI_F_AI_033, XMANAI_F_AI_035, XMANAI_F_AI_033, XMANAI_F_EX_042, XMANAI_F_EX_044

5.3 Feature Assessment

In order to assess the added value that the feature list that has been extracted and defined in section 5.2 brings to the XMANAI demonstrators as stakeholders that represent different manufacturing industries and the AI community in general, a dual assessment has been followed for the internal (within the consortium) MVP feature assessment: (a) Business-related assessment in order to gauge the feedback of the 4 demonstrators (FORD, WHIRLPOOL, CNH, UNIMETRIK), and (b) Technical-related assessment in order to obtain the perspective of all technical partners. Since the MVP definition is considered as a live, continuously evolving learning process that cross-cuts the design-development-demonstration activities across the different WPs, only the internal assessment will be reported in the

context of this deliverable and shall be complemented with the full, generalized assessment in the 2nd release of the WP1 activities on M18.

Upon elaborating on the list of features that may constitute the XMANAI MVP based on the detailed technical requirements, the XMANAI demonstrators were requested to describe and rate online in a qualitative manner: (a) the Added Value of each feature for their organization (how useful and important each feature is for their internal operations); (b) the Innovation in Manufacturing per feature (how innovative and crucial they consider each feature to be for manufacturing in general). The Profiles (Business User, Data Scientist, Data Engineer) that contributed to this assessment should be also added (in line 2 per organization). The scale that has been adopted builds on the proposition of Lant[2] regarding the assessment of the business value and adapts it to the broader context of features (rather than concrete user stories that had been brainstormed in more fine grained detail).

Figure 5-2 presents the aggregated demonstrators' assessment towards the XMANAI MVP. As indicated in the figure, there are many features that are considered as important or very important (scoring above 5) while very few features were considered as trivial (scoring between 1.5 and 2). There were certain discrepancies noticed in the evaluation between the added value for own organization in comparison to the innovation for industry, especially regarding the data security and sharing functionalities.

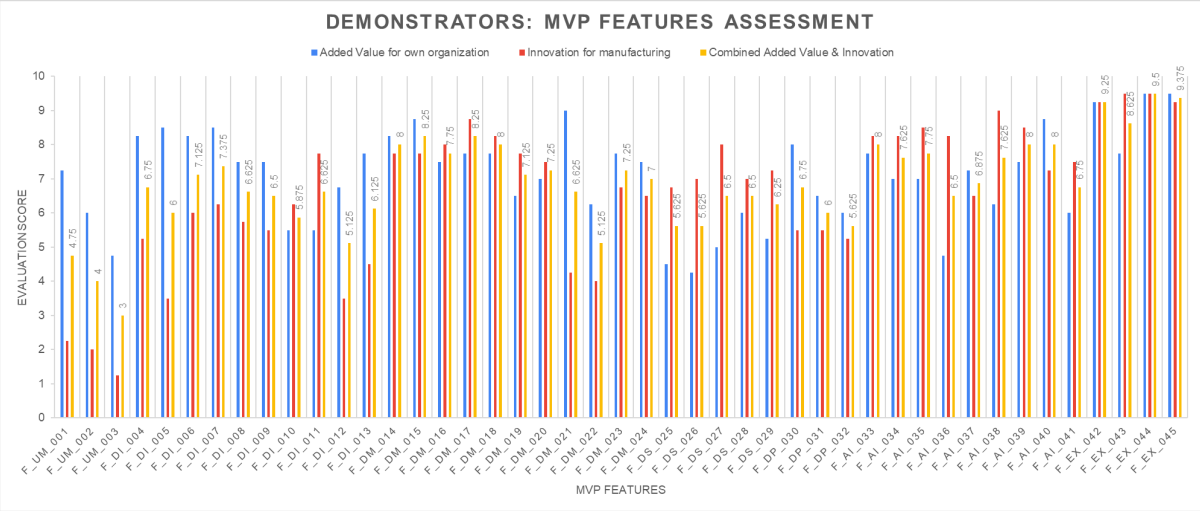


Figure 5-2: XMANAI MVP Feature Assessment – Demonstrators' View

From a technical perspective, the XMANAI technical partners (all partners except the demonstrators) were requested to describe and rate online in a qualitative manner: (a) the Complexity of each feature (how complex and challenging they consider its implementation from a technical perspective); (b) the Feasibility of each feature (how feasible they consider the implementation of each feature in XMANAI – by the end of the project); (c) the Innovation associated to each feature (how novel they consider a feature from a technical perspective). Figure 5-3 presents the aggregated technical assessment towards the XMANAI MVP. As indicated in the figure, there are many features that are considered as highly innovative and feasible, but rather complex in their implementation (e.g. all explainability-related and the AI pipelines features) while there are other features that are considered as feasible and easy to implement, yet they are not really innovative considering the market and the latest scientific advancements.

[2] Michael Lant (2010). How to Easily Prioritize Your Agile Stories. Available at: http://michaellant.com/2010/05/21/how-to-easily-prioritize-your-agile-stories/

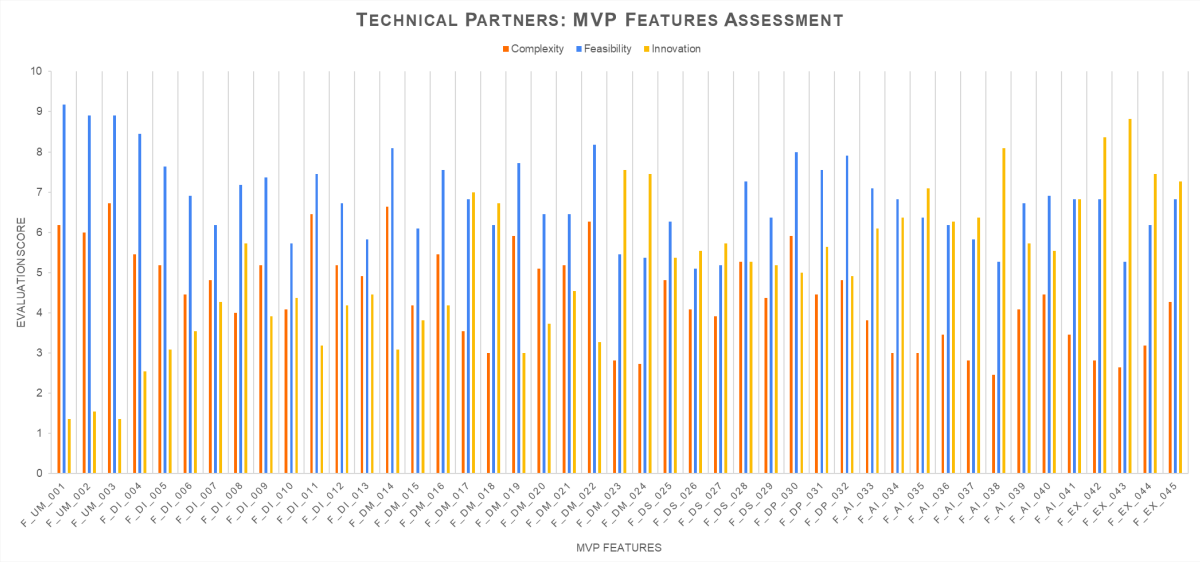


Figure 5-3: XMANAI MVP Feature Assessment – Technical Partners' View

5.4 Draft MVP Consolidation

Taking into account the preliminary assessment reported in section 5.3, the draft XMANAI MVP has been defined based on the combined business and technical assessment (added value versus technical innovation).

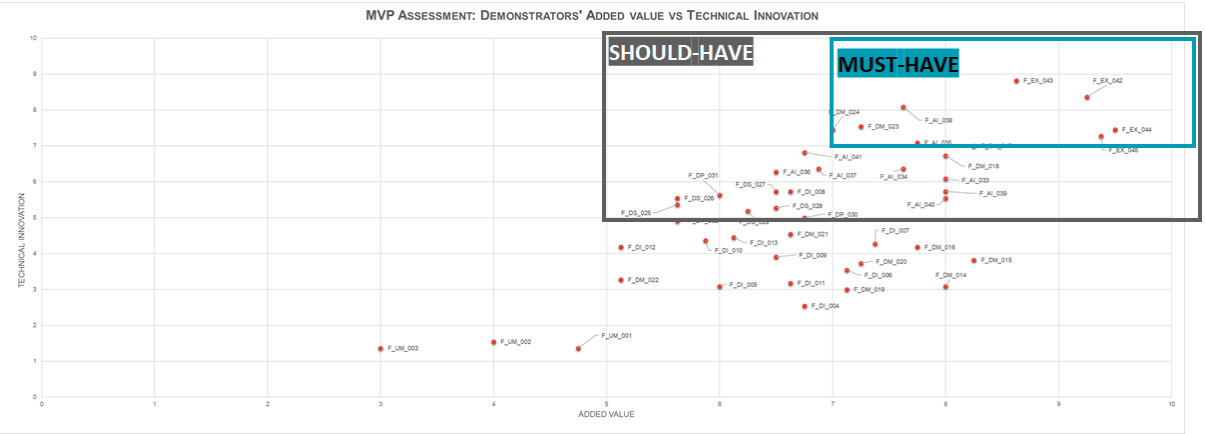


Figure 5-4: XMANAI MVP Feature Combined Assessment

Following the MoSCoW approach (that classifies the requirements into must-have, should-have, could-have, and won't-have, or will not have right now), the set of features that comprise the preliminary XMANAI MVP are depicted in the following table. It needs to be noted that Must-have features have over 7 added value and technical innovation; Should-have features have a combined rank of above 5; Could-have features appear due to their relation to Should-have features, and Won't-have features are low prioritized features (in which the feasibility is also ranked low).

Table 5-1: Preliminary XMANAI MVP

ID	Title	Assessment
XMANAI_F_UM_001	Organization-based access (users within the same organization can access all data assets that belong to an organization)	Could-have
XMANAI_F_UM_002	Project-based access (users from different organizations can access data assets that are grouped under a project)	Won't-have right now

ID	Title	Assessment
XMANAI_F_UM_003	Delete all data assets	Won't-have right now
XMANAI_F_DI_004	Data Sources Management (add/edit/remove/configure/schedule)	Could-have
XMANAI_F_DI_005	Data Secure Uploading as file(s)	Could-have
XMANAI_F_DI_006	Data Secure Uploading via API	Could-have
XMANAI_F_DI_007	Data Mapping to a Data Model and Harmonization (data type casting, transformation to common measurement unit/timestamp)	Could-have
XMANAI_F_DI_008	Data Model Management (view model, add/edit/remove model concepts, versioning different representations)	Won't-have right now
XMANAI_F_DI_009	Data Cleansing (quality checks, cleaning rules definition and execution before storage)	Could-have
XMANAI_F_DI_010	Data Anonymisation (configuration and execution before storage)	Could-have
XMANAI_F_DI_011	Data Storage in central XMANAI Cloud Storage	Could-have
XMANAI_F_DI_012	Data Storage on-premise / in private cloud	Could-have
XMANAI_F_DI_013	Data Encryption (in transfer, in storage)	Won't-have right now
XMANAI_F_DM_014	Dataset Management (add/edit/remove asset and metadata, versioning)	Could-have
XMANAI_F_DM_015	Features Management (store, add/edit/remove asset and metadata, define rules, versioning)	Won't-have right now
XMANAI_F_DM_016	Results Management (edit/remove asset and metadata, versioning)	Could-have
XMANAI_F_DM_017	AI Model Management (create/store/update/delete/clone/export/import, configure, versioning, register/import)	Should-have
XMANAI_F_DM_018	AI Pipeline Management (create/store/update/delete/clone/export/join, view IPR of assets involved, define templates, versioning)	Should-have
XMANAI_F_DM_019	Data Asset Export (download file or via API depending on IPR)	Won't-have right now
XMANAI_F_DM_020	Data Asset Access Policies and Licencing (configuration, management and enforcing)	Should-have
XMANAI_F_DM_021	Data Asset Secure Transfer (through platform layers, operation and enforcing)	Could-have
XMANAI_F_DM_022	Data Asset Access and Activity Logging	Could-have
XMANAI_F_DM_023	AI Model Security Assessment	Must-have
XMANAI_F_DM_024	AI Pipeline Security Assessment	Should-have
XMANAI_F_DS_025	Data Asset Sharing (based on contracts, with selected users / organizations)	Should-have
XMANAI_F_DS_026	Data Asset Trading (based on contracts, payment performed offline/ online)	Should-have
XMANAI_F_DS_027	Data Asset Contract Management (contract preparation, negotiation, agreement, enforcement)	Should-have
XMANAI_F_DS_028	Data Asset Search and Discovery (including indexing, sorting, filtering, matching level to your asset, view details)	Should-have
XMANAI_F_DS_029	Secure Transfer of data assets acquired to their legitimate consumers (across platform layers, history log)	Won't-have right now
XMANAI_F_DP_030	Data View & Visualisation (query data, view distribution, statistics, aggregations, time periods, descriptive analytics, preview sample)	Should-have
XMANAI_F_DP_031	Data Manipulation (merge, split, augment, resample, aggregate, create new features, handle missing values, etc.)	Should-have
XMANAI_F_DP_032	Data Transformation (normalisation, encoding, modifying data types, etc.)	Could-have
XMANAI_F_AI_033	AI Model Design (define, configure, store, import, export)	Should-have
XMANAI_F_AI_034	AI Model Training, Application & Evaluation (experimentation vs production, configure control parameters, define/monitor eval . metrics, save check points, support parameter optimisation)	Should-have
XMANAI_F_AI_035	AI Pipeline Design (define, configure, register AI model, add annotations/comments, reuse common features)	Must-have
XMANAI_F_AI_036	AI Pipeline Execution & Evaluation on XMANAI Common Cloud (experimentation vs production, run automatic tests for AI models, run scheduling, configuration)	Should-have

ID	Title	Assessment
XMANAI_F_AI_037	AI Pipeline Execution & Evaluation on Premise / Private Cloud (experimentation vs production, run automatic tests for AI models, run scheduling, configuration)	Should-have
XMANAI_F_AI_038	Collaboration over AI pipelines creation (experiments comparison, history of events, simulations of different settings, models, and methods for same task)	Must-have
XMANAI_F_AI_039	AI Pipeline Results Management (store, export, add summary statistics, easily compare with real values (for predictions), retrieve via API)	Should-have
XMANAI_F_AI_040	AI Pipeline Results Visualisation (configuration of various charts, add comments, store, export, run on cloud vs on premise)	Should-have
XMANAI_F_AI_041	AI Pipeline Support (offer recommendations, guidelines, common metadata model, execution logs, error handling, show comp. resources expended)	Could-have
XMANAI_F_EX_042	Explainability Methods Management (add/remove/configure, register/import)	Must-have
XMANAI_F_EX_043	Collaboration over AI model/results/pipelines explanations (application of explainability methods at AI pipeline or model level, results querying)	Must-have
XMANAI_F_EX_044	Explainability Results Visualisation (various charts, adjust based on user profile)	Must-have
XMANAI_F_EXI_045	Explainability Results Evaluation (allow manual feedback & results validation)	Must-have

Through the above consolidation of the XMANAI MVP, it is evident that explainability will be pursued in XMANAI in three axes:

- **Understanding Data** as the primer towards AI explainability that can be ensured by properly ingesting data, extracting their structure and semantics, and allowing for sample data exploration, summary statistics and visualizations.
- **Explaining Results of AI models** in a comprehensive, yet interactive way through different explainability techniques in order to bring to the same page both business users and data scientists / engineers.
- **Understanding the inner workings of AI models** in order to build robust and reliable AI solutions that shall inspire trust to the manufacturers.

It needs to be noted that the technical requirements of the backlog presented in Section 3.2 essentially inherit the prioritization of the features to which they are allocated (based on the relations identified in Section 5.2).

As the project advances and the activities towards the elaboration, design and implementation of the different Data and AI Services Bundles proceed, the MVP definition will be continuously revised to reflect the XMANAI progress. All updates performed will be initially documented in D1.3 (due on M18).

6 Conclusions and Next Steps

The present deliverable (D1.2) documents the results of the 1st iteration of tasks T1.3 and T1.4 which have three main objectives: to elaborate on the XMANAI concept (what does XMANAI bring to different stakeholders), to define initial technical and data requirements, and to provide a first version of the MVP (Minimum Viable Product) that will guide the impending design and development activities. To derive these outcomes, a clear and easily comprehensive approach was followed, including: (a) brainstorming of user journeys containing AS-IS and TO-BE scenarios for business users, data scientists and data engineers, (b) extraction of technical requirements and organization into features, (c) business and technical assessment on feature importance, and (d) preliminary MVP definition.

It is worth noting that the XMANAI concept was based on key findings of tasks "T1.1 - Explainable AI and Graph Machine Learning Analytics State-of-Play" and "T1.2-Human Aspects in Decision Making and AI" reported in the XMANAI Deliverable D1.1, as well as on the business requirements presented in the XMANAI Deliverable D6.1. These findings assisted in the agile development of the User Journeys, high-level usage scenarios based on the three representative roles supported by XMANAI: the Business user, the Data Scientist and the Data Engineer. These scenarios revealed the logical flow of information and operations in XMANAI and helped in the elaboration of the TO-BE situations for the different stakeholders. The AS-IS situations, on the other hand, were formed using the information provided by the XMANAI Deliverable D6.1 as the outcome of task "T6.1-Demonstrators Requirements Elicitation".

The next step was to extract and define the technical requirements based on brainstorming in the Miro boards connected to the user journeys. By grouping and organizing such requirements in the backlog, their complementarity with the business requirements and the different steps of the user journeys was revisited.

The final step was to determine the preliminary XMANAI MVP based on a concrete methodology, the high-level user journeys and the business and technical requirements as grouped into features. Based on the description of the different features along with their relation to the technical requirements and their dependencies, the different XMANAI partners proceeded to the MVP feature assessment activities from a technical and business perspective. By consolidating the outcomes, the draft XMANAI MVP designates a set of "must-have", "should-have", "could-have" and "won't-have right now" features.

The MVP definition will be further updated and adapted in the future iterations, yet it needs to be underlined that the XMANAI MVP represents not only a set of features to implement, but also the mentality of work that ensures the XMANAI platform will be appropriately validated by its stakeholders and will deliver the maximum added value, with the lowest possible risk.

Since the T1.3 and T1.4 remain active, the next steps along the proposed work include the take-up of the results by: (a) the XMANAI architecture in task "T5.1- Platform Architecture, Bundles Communication Design and APIs Definition", (b) the Data & AI Services Bundles in WP2 "Industrial Asset Management and Secure Asset Sharing Bundles" and WP3 "Core Artificial Intelligence Bundles for Algorithm Lifecycle Management" since the specified MVP prioritization directly affects their design and development activities, and (c) the AI Models-related tasks of WP4 "Novel Artificial Intelligence Algorithms for Industrial Data Insights Generation" as the data profiling activities and their underlying business problems (also described in D6.1) will guide the initial selection of AI algorithms for the draft XMANAI XAI Models Catalogue. All the D1.2 results, especially the XMANAI MVP and the requirements backlog, will continue to be further reflected and improved while the updates performed will be reported in D1.3 and D1.4 that are due on M18 and M30, respectively.

References

Gerdeman, D., 2017. Companies love big data but lack the strategy to use it effectively. Harv Bus Sch Work Knowl.

Ismail, A., Truong, H.L. and Kastner, W., 2019. Manufacturing process data analysis pipelines: a requirements analysis and survey. Journal of Big Data, 6(1), pp.1-26.

Lenz, J., Wuest, T. and Westkämper, E., 2018. Holistic approach to machine tool data analytics. Journal of manufacturing systems, 48, pp.180-191.

Luke Posey, "Engineering + Data Science: The Missing Duo", 2019. Available at: https://towardsdatascience.com/engineering-data-science-the-ultimate-yet-somehow-missing-duo-597eb21dda98

Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Flach, P., Hernández-Orallo, J., Kull, M., Lachiche, N. and Ramírez-Quintana, M.J., 2017. Casp-dm: Context aware standard process for data mining. arXiv preprint arXiv:1709.09003.

Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Orallo, J.H., Kull, M., Lachiche, N., Quintana, M.J.R. and Flach, P.A., 2019. CRISP-DM twenty years later: From data mining processes to data science trajectories. IEEE Transactions on Knowledge and Data Engineering.

Ries, E., 2009. Venture Hacks interview:" What is the minimum viable product?". URl: http://www. startuplessonslearned. com/2009/03/minimum-viable-product. html(visited on 17/06/2017).

Schmidt, C. and Sun, W.N., 2018. Synthesizing agile and knowledge discovery: case study results. Journal of Computer Information Systems, 58(2), pp.142-150.

XMANAI Description of Action (DoA), 2020.

XMANAI Deliverable D1.1 "State-of-the Art Review in XMANAI Research Domains", 2021.

XMANAI Deliverable D6.1 "Demonstrators Requirements", 2021.

XMANAI Deliverable D9.2 "Ethics and Data Management Plan", 2021.

Wake, B., 2003. Invest in good stories and smart tasks, August 2003. Retrieved from: http://xp123. com/articles/invest-in-good-stories-and-smart-tasks.

List of Acronyms/Abbreviations

Acronym/ Abbreviation	Description
API	Application Programming Interface
BR	Business Requirement
DoA	Description of Action
JSON	JavaScript Object Notation
MVP	Minimum Viable Product
TR	Technical Requirement
WP	Work Package
XML	Extensible Markup Language