CONTRACT FRAMEWORK FOR PREDICTIVE & SMART MAINTENANCE

Introduction

Effective collaboration, illustrated by SAMEN (Smart Maintenance Enabled Business), depends on trust among participants, extending beyond mere contractual obligations. Contracts act as protectors of valuable data, solutions, and knowledge, However, the market seeks a standard contract framework that offers a complete perspective on contracting smart maintenance solutions.

Currently, the legal field faces challenges in keeping up with the rapad advancements of technology. Numerous parties lack the resources or know-how to create agreements that effectively protect their priceless assets. This becomes especially clear when sharing company data in collaborative networks. The difficulty goes beyond just having the capability; it involves negotiating terms and conditions that satisfy all parties involved.

To address this requirement, the Contract Framework presented here introduces essential elements and principles. Its purpose is to ease discussions regarding contracted smart maintenance solutions within the existing legislative landscape, which frequently lags in adapting to current needs.

The development process involved collaborative efforts centered around an initial mind map, providing a structured hierarchy visible in the index. Within this document, notes and links accompany each topic. While the aim is to provide a valuable starting point for parties involved in smart maintenance solution contracts, it's recognized that the document may not over every detail due to the rapidly changing nature of smart solutions.

This compilation represents the outcome of two years dedicated to thoroughly reviewing existing literature and professional documents. While certain information might not be groundbreaking for all readers, considering the swift advancements in smart solutions, the relevance of the provided information might quickly become outdated. It's important to note that legislation is undergoing development, although it lags behind the dynamic pace of the current market.

November 2023,

World Class Maintenance,

Fieldlab SAMEN (Smart Maintenance Enabled)

Authors: Feng Fang, Pedro Durlinger, Ruud Poppelaars

Table of contents

How to read

The Contract Framework is organized based on the BLOFT model, derived from Data Sharing Canvas[1], encompassing Business, Legal, Operations, Functionality, and Technical considerations.

- Business: This aspects involves exploring the business elements and contextual factors that fall within the scope of the contract.
- Legal: By examining existing and applicable legislations related to data and the contracting of smart solutions, this facet ensures legal compliance.
- Operations: This segment outlines critical considerations necessary for operational use and the establishment of reliable solutions.
- Functionality: Addressing the contracted functional requirements and flows, this aspect clarifies what is include and excluded from the agreement.
- Technical: With a focus on the infrastructure required for the successful implementation of solutions, the Technical component ensures the smooth integration of smart maintenance systems.

This document is a valuable resource, offering insights and guidance for individuals engaged in contracting smart maintenance solutions, whether as providers or users, especially in the domain of predictive maintenance.

Readers can start their exploration from the following chapters or use the index links to directly access areas of particular interest. The content aims to provide practical tips and significant considerations, enriching the comprehension of the diverse landscape related to contracting smart maintenance solutions.

Business

The business dimension of engaging in smart maintenance contracts involves various crucial components, including context and objectives, roles, and the fee structure.

Context & Objectives

A thorough smart/predictive maintenance contract must clearly outline the project's context and objectives. This includes providing a detailed description of the assets involved, the value proposition, the desired state, and the organizational principles guiding the initiative.

Clear definitions of assets in scope: The contract should accurately specify the assets encompassed within its scope and those explicitly not included. Additionally, it should clarify the interactions between assets within and outside the scope, guaranteeing a clear division of responsibilities.

Clearly communication benefits: The value proposition plays a crucial role in explaining the advantanges that customers gain from participating in a smart/predictive maintenance project. This encompass economic benefits such as cost savings or increased revenue, social advantages linked to fostering a positive brand image in sustainability and digitalization, and operational benefits such as improved performance.

Clear description of desired state: The contract needs to clearly outline the goals for the smart/predictive maintenance project/program. This requires a comprehensive description of the intended condition of the assets. It’s important to consider how adaptable these goals are, establish a roadmap for achieving them, and acknowledge possible differences in how partners interpret these objectives. This helps ensure a common understanding and agreement on the main goals of the initiative.

Clear communication of organizational principles: When establishing organizational principles in a contract, clearly communicating core values is essential. Two examples of such principles are agile and compliance principles. Agility as core principle involves incorporating factors like aligning both externally and internally. Additionally, prioritizing the readiness for is crucial, showing a dedication to adaptability when facing evolving situations. This guarantees a collaborative environment that is dynamic and responsive. Another vital aspect is compliance principles, covering elements such as adhering to digital security policies. Moreover, committing to legal compliance concerning intellectual property rights ensures a secure and legally sound foundation for collaboration. Clearly outlining these principles promotes a shared understanding, fostering a collaborative atmosphere grounded in mutual respect and shared values.

Roles

Within the framework of smart maintenance contracts, various key roles have been identified, including:

Algorithm Owner: The party or parties that, through contractual agreements, claim the right to a specified algorithm, governing its usage and application.

Algorithm Developer: The party or parties that, through contractual arrangements, hold exclusive rights to a defined test, training, and development model that forms the basis for the algorithm's (further) evolution, encompassing data science, neural networks, etc.

Test Data User: The party or parties that, through contractual agreements, borrow specified contextual data to create a test set, aiding the Algorithm Developer in training the model.

Data Owner: The party or parties that, through contractual stipulations, possess sole ownership of specified contextual data used for model testing and streaming data.

Asset Manager: The party responsible, through contractual commitments, for making decisions concerning asset maintenance within the predictive maintenance contract.

Asset Owner: The party serving as the financial and legal owner of assets covered by the predictive maintenance contract.

Platform Supplier: The party that, through contractual agreements, provides the platform enabling the predictive maintenance algorithm to function by delivering streaming data and actionable messages (prediction results). This platform could be a SaaS or third-party solution.

Data Accessor: The party that, through contractual arrangements, extracts streaming data from assets within the predictive maintenance contract on behalf of the Data Owner and delivers it to the platforms of the "Platform Supplier." This party is also responsible for conveying action-oriented messages from the "Platform Supplier" to the relevant parties.

Service Provider: A party contracted separately by the Asset Manager or Asset Owner, receiving actionable messages to be used in accordance with the agreed-upon services and conditions.

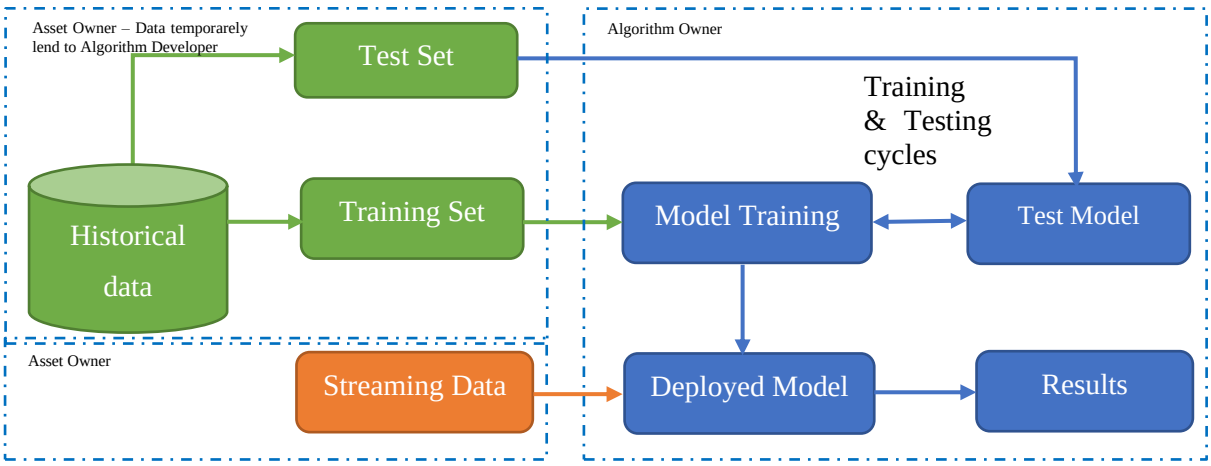

Figure 1 on main roles within predictive maintenance

Fee Structure:

The fee structure is a vital component within any smart/predictive maintenance contract, covering compensation methods and financial arrangement.

Compensation Mechanisms: Various compensation mechanisms play a pivotal role in structuring the financial aspects of the contract. Key types [2] include:

- Volume Contract: Often involves a fixed price based on a predetermined number of activities or services.
- Effort Contract: Focuses on compensation for the expended effort, ensuring payment aligns with the resources utilized.
- Result Contract: Hinges on the achieved outcomes, where compensation is contingent upon meeting predefined results.
- Performance Contract: Tied to the overall performance, this method ensures compensation aligns with the agreed-upon performance metrics.

As-a-Service: Reflecting a subscription-based model where compensation is structured as a service fee for ongoing provision.Understanding and choosing the appropriate compensation mechanism is pivotal in establishing a fair and effective financial framework for the smart/predictive maintenance contract.

Legal

In the legal aspect of contracting for smart maintenance, our initial exploration focuses on fundamental laws and legislations. Specifically, we identify two critical legal frameworks: intellectual property law concerning artificial intelligence and legislation related to digitalization. Subsequently, we carefully examine legal contracts that oversee data ownership and governance.

Intellectual Property laws for artificial intelligence

Artificial Intelligence (AI) systems traditionally find protection under three pillars of Intellectual Property (IP) frameworks: copyright, patent, and trade secrets laws. Copyright is an insufficient avenue due to algorithms' exclusion from protection under the EU Software Directive [3]. This approach faces challenges in meeting the criterion of author's "intellectual creation" and doesn't always cover the creative expression of such creations. As a result, rights holders currently opt to protect these mathematical tools through trade secrecy privileges provided by the EU Trade Secrets Directive (EUTSD), a practice that involves significant risks. Trade secret laws were not originally intended for IP protection, and many EU Member States do not consider trade secrets a part of the IP domain. A more effective approach is to acknowledge that patents are specifically designed to protect technical inventions. Although still evolving, the protection of algorithms under patent law is expanding, with increasing recognition of computer-implemented inventions (CIIs). This path stands as more preferable and logical choice, offering the advantage of avoiding the drawbacks associated with trade secrets law while acknowledging the innovative nature of AI.

Intellectual property law for database

Legally, a database is defined as a collection of works, data, or other distinct elements that are organized in a systematic or methodical manner and can be accessed electronically or separately by other means. This database receives automatic protection under the Database Law for a specific duration, subject to meeting various conditions. Thus, the Database Law safeguards the database and its content [4] [5] [6].

Data ownership

No law yet on ownership: Ownership typically pertains to a tangible object by principle. Legally, a tangible object is defined as something physically graspable or subject to human control. As data lacks physical touchability, ownership of data cannot be aserted.

Protecting data as trade secrets: The Trade Secrets Protection Act (TSPA) provides specific legal rights to rightful holders of trade secrets, empowering them to take legal action against infringements upon these secrets. For instance, they may pursue an injunction to prevent the use of trade secrets. To initiate a claim under the TSPA, several conditions must be satisfied, which involve requirements related to confidentiality (such as NDAs).

Use case from Equans on how to protect usage right: Ownership of data, such as a lists of names, IP addresses, or sensor data, is not a recognized concept. Each individual data point lacks inherent value. Only when combined with other data do they attain significance, creating information that can be valuable, based on the quality and quantity of data within the dataset. As data ownership is nonexistent, establishing contractual agreements regarding its utilization becomes crucial. This may involve aspects such as specifying usage rights. Within the context of the contractually agreed-upon services, party X holds the right of use, free of charge, of the work-relevant data obtained from or generated by the client. This data may encompass information sourced from databases, such as a maintenance information system or sensor data, among other possibilities. These usage rights extend throughout the entire duration of the agreement. Upon termination fo the agreement, any copies of the provided data must be destroyed and/or returned within X months.

Legal Contract

No contract templates are currently available in the predictive maintenance domain. Nevertheless, there are publicly accessible examples related to similar topics. Here we aim to illustrate what the core contractual elements in a smart maintenance contract. Example 1 shows a contract template for data sharing [7]: FME Advocaten has drafted a free collaboration agreement, 'Dare-2-Share,' with input from, among others, TNO. Entrepreneurs can use this agreement to establish fair and reliable terms in the 'collaboration in innovation' phase, where data is shared between large and small companies. Example 2 shows general contract elements in a service contracts [8] . Example 3 shows a data-driven asset management in practice [9]. Experience has been gained in a number of pilots with wet structures, movable bridges and tunnels, in which management and maintenance have been tested at national and regional level using current data, artificial intelligence and dashboarding. The key question was: 'does it work?'. The next period will focus on the 'standardization phase'. In this phase, Rijkswaterstaat creates the organizational, process and technical preconditions for data-driven asset management within the entire organization for wet structures, movable bridges and tunnels. The key question in this phase is: 'how do we as Rijkswaterstaat work with it?' By sharing existing data about Rijkswaterstaat's assets and using new data sources, Rijkswaterstaat and 5 managing contractors want to work together towards better data-driven asset management. Rijkswaterstaat and the companies Istimewa, BAM Infra, Heijmans, SPIE and Vialis have recorded this in a cooperation agreement [10], which was signed at the InfraTech trade fair.

Other legislation on data

Data governance act: The Data Governance Act (DGA) [11] [12] is a comprehensive cross-sectoral tool designed to enhance data availability. It achieves this by regulating the re-use of publicly or privately held protected data, promoting data sharing through the oversight of innovative data intermediaries, and incentivizing data sharing for altruistic purposes. The DGA encompasses both personal and non-

personal data, and when personal data is involved, it is subject to compliance with the General Data Protection Regulation (GDPR). Furthermore, the inclusion of built-in safeguards serves to bolster trust in data sharing and re-use, which is essential for expanding the availability of data in the market.

Technical

Regarding the technical aspect within the contract framework for smart maintenance, our attention is directed towards crucial elements such as information security, platform architecture, data principles, and the complexities associated with data sharing.

Information security

One of important aspects of technical principles is the information security. A widely used model to explain information security is the CIA triad, which covers Confidentiality, Integrity and Availability[1].

- Confidentiality aims to ensure that only authorized individuals can access and modify data, typically achieved through secure access controls.
- Integrity ensures that data remains in the correct state, often maintained through methods like digital signatures, hash algorithms, and cryptography.
- Availability, facilitated by specific high availability protocols and network architecture, guarantees timely and reliable access to data for authorized users.

Confidentiality:

The "Secure by design" approach stands as the most effective method to uphold confidentiality by intricately integrating security into the design and architecture of software product. This proactive strategy involves early consideration of various security strategies, tactics, and patterns during the design phase, leading to inherently secure software. The Cybersecurity and Infrastructure Security Agency (CISA) acknowledges the pivotal role of the secure by design approach in embedding cybersecurity into the development of technological products.

Implementing secure by design principles during the design phase significantly reduces exploitable flaws before the product reaches the market. Depending on business's certification requirements, multiple systems, such as ISO9001, ISO27001, and IEC62443, ensure compliance with industry standards while addressing specific security considerations.

To ensure authenticity, one effective method is EBIOS[13], a renowned French reference approach that aids organizations in identifying and comprehending their digital risks. EBIOS facilitates the identification of security controls tailored to specific threats, establishing a framework for monitoring and continuous improvement based on a comprehensive risk analysis shared at the highest organizational level. This method not only assists in recognizing potential risks but also enables the strategic implementation of measures that align with the identified threats, fostering a proactive and adaptive approach to cybersecurity.

Data Integrity:

Data integrity refers to the accuracy and consistency of data across its entire life cycle – from capture and storage to processing, analysis, and utilization. Managing data integrity involves ensuring that data remains complete and accurate, free from errors or anomalies that could compromise its quality. Well-recorded and stored data, maintained accurately and consistently, preserves integrity. Conversely, distorted or corrupted data becomes unreliable for business purpose.

In regulated industries where data accuracy, completeness, and verifiability are crucial, data integrity management holds significant importance. Poor data integrity can result in financial losses, harm to public and industrial reputations, and significant disruptions to production schedules. Therefore, maintaining robust data integrity is crucial for sustaining operational reliability and trustworthiness.

Key characteristics of data integrity include:

- ***Accuracy:*** Data must be error-free and accurately reflect the real-world scenarios or events to prevent erroneous analyses and decisions.
- ***Consistency:*** Data should remain constant across all instances and time, changing only when intentionally updated or modified.
- ***Completeness:*** Comprehensivee data includes all necessary components and information, facilitating precise conclusions and decision-making processes.
- ***Reliability:*** Reliable data is trustworthy in terms of accuracy and consistency, bolstering critical decision-making.
- ***Timeliness:*** The availability of data at the required time for decision-making is crucial; delayed data can be as detrimental as inaccurate or incomplete data.
- ***Validity:*** Valid data conforms to predetermined formats and values established during the data design phase, ensuring suitability for specified purposes.

Data Availability:

The primary challenges that affect data availability include:

- ***Host server failure:*** When the server storing data encounters a failure, the data becomes inaccessible.
- ***Data quality:*** Redundant, inconsistent, or incomplete data presents a challenge as it may not be useful for IT operations.
- ***Legacy data:*** Outdated data can become obsolete and, consequently, unusable.
- ***Storage failure:*** If the physical storage device fails, the data stored on it becomes unavailable.
- ***Network crashes:*** Failures within the network can make any data accessed through it unavailable.

- ***Slow data transfers:*** Data transfer speed may suffer based on the location of data storage and its usage.
- ***Data compatibility:*** Data functional in one environment may not align with the requirements of another.
- ***Security and data breaches:*** Unauthorized access by malicious actors can compromise and block an organization's data, as seen in incidents such as ransomware attacks.

Efficient best practices for addressing challenges related to data availability include:

- ***Redundancy and backups:*** Regularly backing up data stands as a crucial measure in ensuring availability. Storing backups in separate locations or a distributed network helps to mitigate risks.
- ***Data loss prevention tools:*** Implementing tools specifically designed to prevent data loss fortifies the overall security posture, offering protection against potential threats.
- ***Erasure coding:*** This data protection method involves fragmenting data, expanding it, encoding it with redundant pieces, and dispersing it across various locations or storage devices, ensuring resilience.
- ***Retention policies and procedures:*** Adhering to clearly defined retention policies ensures that unnecessary data or devices are securely archived or properly disposed of, minimizing risks.
- ***Automatic switch to backups:*** Introducing flexibility through automatic switching to backup or failover environments in event of drive failures or data losses, ensures a seamless transition and maintains continuous data availability.

Platform architecture:

Platform is a critical element in the technical dimension of a contracting framework. Here are key tips for a successful platform team according to Business Trust Architectuur [14]:

- ***Establish a Clear Mission and Role:*** Define a clear mission tailored to the team's objectives right from the outset.
- ***Treat Platform as a Product:*** Embrace a product-oriented mindset that prioritizes delivering tangible value to internal customers, particularly app developers, based on their feedback.
- ***Address Common Problems:*** Identify and resolve shared issues within the organization, starting by understanding the pain points and areas of friction for developers.
- ***Acknowledge the Value:*** Combat the perception of platform teams being solely cost centers by internally promoting and demonstrating their value proposition.

- ***Avoid Duplication:*** Prevent teams from duplicatring solutions to common problems, thus steering clear of repeating the same pitfalls within the platform team.
- ***Ambrace Flexibility, Speed, and Pay-Per-Use:*** Utilize cloud-based computing to establish a modern, adaptable data platform that is swift to set up and offers flexible, pay-per-use pricing, catering to evolving user needs over the long term.

Data Principles

Data principles provide guidelines [1] for organizations to improve usefulness of digital assets. One wildly used model is FAIR data principles which aim to improve the findability, accessibility, interoperability and reusability (FAIR) of data.

The FAIR principles emphasize machine-actionability (i.e., the capacity of computational systems to find, access, interoperate, and reuse data with none or minimal human intervention) because humans increasingly rely on computational support to deal with data as a result of the increase in volume, complexity, and creation speed of data.

FAIR data principles:

Findable: the first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers.

Accessible: once the user finds the required data, they need to know how they can be accessed, possibly including authentication and authorisation.

Interoperable: the data usually needs to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

Reusable: the ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

There are different types of data: closed data (e.g., context data about primary process and market), real-time data collected from sensors, combined data (context data about predictive maintenance and other maintenance)

- ***Closed Data*** is defined as “data that can only be accessed by its subject, owner or holder.” Context data regarding the primary process provides background information related to significant data points. It constitutes the environment in which all your other data observations occur. For instance, process plants generate extensive time series data daily from sensors, monitors, and smart devices. Context data related to the market (public) is general accessible data, often freely available for use.
- ***Real-time Processing:*** This involves handling streams of data captured in real-time, processed with minimal latency to generate real-time or near-real-time reports or

automated responses. It deals with the processing of an unbounded stream of input data, requiring very short latency for processing — typically measured in milliseconds or seconds.

- ***Real-time (Sensor) Data:*** Combining sensor data with proces, maintenance, failure and general public data can further refine predictive maintenance strategies.
- ***Logging:*** Participants logging is necessary for reporting purposes to ensure accountability. A transparent enrollment process should also be specified with a focus on audit trails, system logging and reporting requirements.

Data sharing

In this section, we address technical aspects to facilitate data sharing [1], including data exchange, data protocol/standards, message format, and data formatting.

- ***Data Exchange:*** Prior to standardization and implementation of an exchange protocol, it's essential to define functional requirements for data exchange. These requirements should align with the specific data transfer characteristics of various use cases. For example, APIs may suffice for transferring small amounts of data, whereas large data volumes may necessitate alternatives like an FTP server. Service providers increasingly offer ready-to-run, pay-as-you-use solutions, such as data service hubs.
- ***Protocols / Standards:*** Data standards establish a shared understanding of data semantics, structure, and formatting. The data exchange protocol, comprising formal rules, regulates the format, timing, sequencing, and error control in data exchange among parties.
- ***Message Format:*** Also referred to as message standards or formats, these specifications outline how data/documents are presented using specific syntax. Examples include XML-based message "standards" like UBL (Universal Business Language) and XBRL (eXtensible Business Reporting Language), designed for business documents previously transmitted in paper form. Various industries often have their own standards, such as papiNet (forest and paper), RosettaNet (ICT industry), and HL7 (healthcare), catering to specific vertical business needs.
- ***Data Formatting:*** Data formatting involves tailoring the appearance of data in a document without modifying its content.

Operational

In addressing the operational dimensions of the contract framework for smart maintenance, we delve into operational governance, incident management, and tooling.

Operational Governance

A fundamental aspect of establishing trust involves defining precise expectations and requirements throughout the entire contracting process. It also involves ensuring compliance with these requirements by all involved parties. This encompasses fostering transparency across all stages of contracting: from the preliminary stages before Data Sharing through Trust Framework agreements, during the actual Data Sharing via Data Service Transaction Agreements, amd after data sharing through Dispute Management.

Contract (Framwork) elements are:

Dispute Management Process

A Dispute occurs when actors within the Trust Framework cannot settle disagreements between them. Dispute Management is the process for managing all reported Disputes.

SLAs for the process of solving disputes

Disputes (between Data Service Providers and Data Service Consumers) should be supported to the Trust Framework Authority for cross-Domain Data Sharing.

SLAs for the analyse of reported disputes

The Trust Framework should define service level agreements for the process of solving disagreements to clearly define when a disagreement becomes a Dispute (for example “max solvingtime”). Determine the need and extent of an appeal process

Minimum Logging requirements

Logging is required to enable actors to be able to provide proof of adhering to various requirements.

ITIL management / Security management

The processes incident management, problem management, configuration management, change management and release management must be offered within a contract (framework).

A number of common ISO/IEC guidelines are listed below:

- ISO/IEC 20000 [15] shows service management system requirements of information technology.
- ISO/IEC 27002 [16] shows guidelines for Information security, cybersecurity and privacy protection.
- ISO 22301 [17] shows requirements for business continuity management systems.

Tooling

Contracting smart and predictive maintenance is usually about value, data exchange, smart soluitons and tooling. Here we state some “backbone” applications to have in place and should be applied for resilience of the solution(s), that can be easily forgotten.

Document Management System (DMS)

A Document Management System (DMS) encompasses a wide range of content, including reference cards, processes, roadmaps, training documentation, news, mailing updates, knowledge bases, API descriptions, API's, life cycle management, etc

Notification platform

Notifications are alerts that inform you about new predictions, updates, or outage events. The way notifications work can vary depending on the device and solution you are using. Inform all parties what available notification strategies are available from the offered solution: Mobile Push Notifications, Web Push Notifications, Email, In-App messages, SMS, Journeys (messages channels) [18].

Test-tooling/scripting

Smart-/Predictive maintenance uses software. Software needs updates, bug fixes and proper software testing before you apply it on you operations. Software testing can provide objective, independent information about the quality of software and risk of its failure to users or sponsors. What test scripts and procedures are in place, are users involved, etc.

Software libraries

Programming libraries [19] are useful tools that can make a developer's job more efficient. They provide pre-written, reusable chunks of code that developers can use to create applications quickly and easily.

Issue-tracker

An issue-trackers [20] is a program to log and track issues within the application. Issue tracking is the process of monitoring problems that users are experiencing with a product or service. Issue tracking software is a tool that records and updates those problems as tickets or cases. Issue tracking can be used for software development, customer service, and other business areas that require collaboration and feedback.

Functional

In the domain of data-driven smart maintenance solutions, significance is attributed to various aspects. These include functional scope, as-a-service functionalities, functional components, security features (e.g., authorization and authentication), and functionalities pertaining to data quality [1]. In additions, we find interation model, user experience and privacy are also important functional aspects when contracting smart maintenance.

Functional scope

The functional scope of the solution defines the core features and functions that the project team must deliver for the respective product(s). It holds signifant important in estimating costs during the project's funding phase and in constructing the product backlog during the elaboration stage. In specific scenarios, the functional scope can serve as a replacement for business requirements. In such cases, analysts are tasked with accurately expressing the functional scope instead of business requirements, possibly bypassing the need for a Business Requirements Document (BRD) process altogether [21].

As a service functionalities

The Operational chapter elucidates the concept of Service Level Agreements (SLAs). It’s crucial to explicitly delineate services such as 24/7 support, training, backup, recovery, restoration, analysis, development, requests, repair, maintenance, and other related options within the agreement and associated requirements. "As a Service (aaS)" denotes a business model delivering services to a customer, whether internal or external. These offerings usually furnish API-driven endpoints and are accessible through a web console in a user's browser. The term "XaaS" encompasses the broader concept of "anything as a service." However, when considering XaaS, careful attention should be paid to vendor-specific lock-in features, which might be more constrictive compared to alternative open-source options [22].

Functional components

The essential functional elements of a software application consist of data storage, data access logic, application logic, and presentation logic [23]. When acquiring a predictive maintenance solution, it’s vital to meticulously evaluate both the included and excluded functional components. This evaluation encompasses various considerations such as model training, model and asset types, lagging analysis, anomaly detection, predictive and prescriptive capabilities (including actionable actions), historical data storage, model development, model deployment, data transfer and streaming, security measures, help functions, visualization, actionable data (notifications, alarms), logging, and other aspects.

Authentication

Authentication and Multi-Factor Authentication (MFA): To validate claimed identities and reduce fraudulent use, actors' identities need authentication. Data Service Providers can specify the level of assurance (LoA) required from their Data Service Consumers, particularly when consumers belong to different domains. Implementing Multi-factor Authentication (MFA) can bolster security without operational complexities. It is crucial to align policies and offer adaptable authentication factors when introducing new solutions and platforms [24].

Regulations on Electronic Identity (eIDAS): In the European Economic Area (EEA) countries, governments commonly identify individuals through their passports, promoting cross-border identity usage. However, electronic identification (eID), like DigiD, often lacks this cross-border recognition. The eIDAS Regulation, established in 2018, facilitates cross-border utilization of electronic identity methods and trust services among EU member states. Once a national eID scheme is notified to the European Commission, it becomes applicable in other EEA countries, enabling individuals to access online services across borders.

Authorisation and Collaboration: The successful implementation of smart maintenance solutions requires collaboration to define roles, establish common terminology, and outline data-sharing agreements. For effective Authorisation decisions by Data Service Providers regarding consumers in diverse domains, essential information must be communicated and aligned with the provider's language and definitions. Authorisation should always originate from the Entitled Party [1].

Functional requirements for data sharing

In the functional domain of data sharing, it's essential to acknowledge the distinct functional needs linked to different data-sharing technologies like Blockchain, Big Data, IoT (Internet of Things), API (Application Programming Interface), and Cloud Computing. When dealing with these technologies, it's crucial to adhere to your company's policies and determine the most suitable data-sharing technology for the specific context of data sharing.

Data quality requirements

Data quality [25] refers to the condition, whether qualitative or quantitative, of pieces of information. According to Wikipedia, data achieves high quality when it is 'fit' for its intended uses in operations, decision-making, and planning, accurately representing the real-world construct it references. With an increasing number of data sources, ensuring internal data consistency becomes crucial, regardless of it suitability for a specific external purpose. Discrepancies in perspectives regarding data quality are commonplace, even when discussing the same data for identical purposes. In such scenarios, data governance is employed to establish consensus on definitions and standards, often involving data

cleansing and standardization to uphold data quality. Establishing clear definitions of data quality and data governance is imperative before embarking on, or collaborating on, smart maintenance solutions.

Access persistency

Persistence is a technique employed to sustain a connection with target systems, ensuring continued access even in the event of machine reboots, shutdowns, or other disruptions [26]. Every platform for smart maintenance solution requires regular maintenance and occasional shutdowns. The platform's built-in functionality and security measures guarantee uninterrupted access during and after hardware maintenance or shutdowns (such as server maintenance). The significance of your primary core processes associated with the solution emphasizes the necessity of establishing suitable contractual terms and incentives.

Interaction model

An interaction model of a smart maintenance platform serves as the foundational structure or blueprint dictating how the product or system behaves, rooted in known user behavior [27]. Two important aspects are functional flow and data flow :

- ***Functional Flow:*** Outline the flow-states within the smart maintenance solution. Evaluate whether the offered processes align with the required needs, encompassing tasks such as anomaly detection, analysis, algorithm enhancement, visualization, decision-making, and more.
- ***Data Flow:*** Define or research the data's collection, transfer, and representation to stakeholders. Assess whether these processes meet the intended needs effectively.

User experience

The user experience encapsulates how an individual interacts with and perceives a product, system, or service, encompassing their impressions of utility, ease of use, and efficiency [28]. In smart maintenance solutions, much like the differences between IOS and Android, functionality and user experience can significantly differy. Although both platforms serve the same purpose, the user experience varies considerably. Two crucial elements in this context:

- ***User experience standardization:*** Caution should be exercised with custom-made or unique solutions. Lack of standardization in screens, buttons, and interactions can lead to inconsistencies and increased development time, as standard UX libraries cannot be utilized for new functionalities.
- ***Channels (Web/Mobile/...):*** Consider the diversity of users and their devices, such as tablets, smartphones, laptops, etc. Evaluate the solution's support for various channels to accommodate different user preferences and needs.

References

[1] Van Beek, R., Willem, E., Dexes, B., Bodde, E., Netbeheer, E. E., Tom Van Engers, N., & Gommans, L. (2021). *Data Sharing Canvas*. https://datasharingcoalition.eu/data-sharing-canvas-2/

[2] K. Smit. (2012). *Technisch Systeem Management: een functie- en informatiemodel voor het beheer, onderhoud en ontwerp van technische bedrijfsmiddelen*.

[3] Katarina Foss-Solbrekk. (2021). Three routes to protecting AI systems and their algorithms under IP law: The good, the bad and the ugly. *Journal of Intellectual Property Law & Practice*, *16*(3), 247–258.

[4] *Databankenrecht*. (2011). https://www.rvo.nl/onderwerpen/octrooien-ofwel-patenten/vormen-bescherming/databankenrecht

[5] *Databankenwet*. (1999). https://wetten.overheid.nl/BWBR0010591/2021-06-07

[6] *De basis van intellectueel eigendom*. (n.d.). Retrieved November 27, 2023, from https://www.rvo.nl/sites/default/files/2023-10/De_basis_van_intellectueel_eigendom.pdf

[7] Samenwerkingsovereenkomst. (2017). https://f.hubspotusercontent20.net/hubfs/7142023/dare_2_share_samenwerkingsovereenkomst_2017_01.docx

[8] Enhancing service contracts to minimize dispute risks. (2017). https://www.dllgroup.com/en/-/media/Project/Dll/Global/Documents/Servitization/EnhancingservicecontractsEN.pdf

[9] Data gedreven Asset management. https://rwsinnoveert.nl/focuspunten/data-iv/@211193/vitale-assets/

[10] Intentieovereenkomst datagedreven assetmanagement. (2023). https://rwsinnoveert.nl/publish/pages/161968/intentieovereenkomst-datagedreven-assetmanagement.pdf

[11] Data governance act explained. (2023). https://digital-strategy.ec.europa.eu/en/policies/data-governance-act-explained

[12] Data governance act. (2022). https://eur-lex.europa.eu/legal-content/NL/TXT/PDF/?uri=CELEX:32022R0868&from=EN

[13] EBIOS method. (2023). https://www.c-risk.com/en/blog/ebios-method/

[14] Business Trust Architectuur. (2023).

https://www.youtube.com/watch?v=yn2WHeEK3QU&t=101s

[15] Service management system requirements. (2018) https://www.iso.org/standard/70636.html

[16] Information security, cybersecurity and privacy protection. (2022). https://www.iso.org/standard/75652.html

[17] Security and resilience. (2019). https://www.iso.org/standard/75106.html

[18] Notification service. (2011). https://en.wikipedia.org/wiki/Notification_service

[19] Software libraries. (2021). https://nl.wikipedia.org/wiki/Bibliotheek_(informatica)

[20] Issue tracker. (2023). https://nl.wikipedia.org/wiki/Iyoyssue_tracker

[21] Functional scope. https://bablocks.com/functional-scope/#:~:text=A%20list%20of%20screens%2C%20features,of%20scope%20for%20the%20project

[22] As a service. (2023). https://en.wikipedia.org/wiki/As_a_service

[23] Forum. (2023). https://brainly.in/question/3438206#:~:text=The%20four%20primary%20functional%20components%20of%20a%20software%20application%20are,application%20logic%20and%20presentation%20logic.

[24] OKTA. (2023). https://www.okta.com/identity-101/why-mfa-is-everywhere/#:~:text=MFA%20secures%20the%20environment%2C%20the,requiring%20additional%20factors%20when%20necessary.

[25] Data quality. (2023). https://en.wikipedia.org/wiki/Data_quality

[26] Linuxforum. (2023). https://infosecwriteups.com/persistence-backdoor-techniques-beginner-to-advanced-in-linux-dd7e109ceeb9

[27] Built-in. (2023). https://builtin.com/design-ux/interaction-model

[28] User Experience. (2023). https://en.wikipedia.org/wiki/User_experience