

Master of Science Thesis

MIOM05 Degree Project in Production Management

Design and Development of a Digital RAM Tool for AUVs

Lund University

Faculty of Engineering LTH

Department of Industrial Management and Logistics

Division of Production Management

Authors: Jacob Persson & Oscar Cedell

Supervisors: Johan Marklund at LTH, Gustav
Rehnborg & Niklas Blomgren at Saab Naval



ABSTRACT

Title: Design and Development of a Digital RAM Tool for AUVs

Authors: Oscar Cedell and Jacob Persson

Supervisors: Johan Marklund (LTH), Gustav Rehnborg (Saab Naval), and Niklas Blomgren (Saab Naval)

Problem description: Autonomous underwater vehicles (AUVs) are long-life, mission-critical systems for which life cycle cost is strongly affected by Reliability, Availability, and Maintainability (RAM). In industrial practice, RAM-related analyses are used to support decisions regarding design, maintenance planning, life extension, and phase-out. Such analyses require systematic collection, structuring, and follow-up of operational and maintenance data over time, often through Failure Reporting, Analysis and Corrective Action System (FRACAS)-related practices and other structured support systems. Saab Naval currently lacks a coherent digital RAM tool adapted to AUVs and their operational availability. Existing commercial alternatives are difficult to apply in this context due to security-classified, export-controlled, and customer-sensitive data, which creates a need for a tailored solution that can operate within secure IT environments.

Purpose: The purpose of the thesis is to develop a digital RAM tool tailored for AUVs. The tool will be based on an analysis of how in-service data feedback can be collected, structured, and presented to support evaluation of RAM.

Method: The study follows a structured development lifecycle, beginning with a requirements elicitation phase informed by a literature review, stakeholder interviews, and benchmarking. These findings serve as the foundation for subsequent requirements engineering and software development. The literature review establishes the theoretical foundation through RAM theory, FRACAS, reliability growth, Integrated Logistics Support (ILS), life cycle management, and the S5000F in-service data feedback specification. Stakeholder input is

collected through semi-structured interviews and continuous dialogue with relevant stakeholders at Saab Naval, including RAM/ILS experts and potential users involved in both data input and analytical use of RAM analyses. The findings are compiled into a product requirements document containing both functional and non-functional requirements, which then guides the design and development of the digital tool. The resulting solution is developed as a web-based application intended for handover and further instantiation within Saab Naval's secure IT environment.

Conclusions: The thesis shows that a structured digital RAM tool tailored for AUVs can be developed by combining established RAM/FRACAS methodology with a pragmatic interpretation of S5000F and the constraints of a secure organisational environment. The web-based tool that was developed demonstrates how problem reporting can be structured and reused to support RAM-related follow-up over time. The work also indicates that long-term value depends on more than the implemented software and calculations. Consistent reporting practices, clear roles and access control, and sustained data quality management are required for the tool to become effective for decision-support. Full validation of the benefits remains dependent on deployment and integration within Saab Naval's secure IT environment and evaluation in live operations.

Contribution: This thesis contributes a structured and documented approach to applying RAM in AUVs. It also contributes with an analysis of how established RAM methods and standards can be adapted to a security-critical industrial context, and with practical experience from developing a digital decision-support tool for use in secure IT environments.

Keywords: Reliability, Availability, Maintainability, RAM, FRACAS, Autonomous Underwater Vehicles, AUV, S5000F

PREFACE

This thesis was carried out as the final part of the Master of Science programme in Industrial Engineering and Management at Lund University. The work was conducted at the Division of Production Management in collaboration with Saab Naval.

We wish to express our appreciation to Niklas Blomgren and Gustav Rehnborg at Saab Naval for their support, commitment, and expert input during the course of the thesis. Their knowledge of both the technical context and the practical needs of the organisation has been highly important to this work.

In addition, we are grateful to everyone at Saab Naval who took part in interviews and shared their experiences, reflections, and perspectives. Their contributions have been essential in building the practical foundation of the study and in making sure that the proposed solution is relevant to the organisation's needs.

We would also like to give special thanks to Johan Marklund at Lund University for his guidance throughout the project and for the valuable discussions that helped shape the direction of the study.

Lastly, we hope that the developed tool can serve as a valuable starting point for continued work within Saab Naval and that it may be further developed and applied within the organisation in the future.

Lund, May 2026

Oscar Cedell and Jacob Persson

TABLE OF CONTENTS

1	Introduction	1
1.1	Background	1
1.2	Organisational Context	3
1.2.1	Saab Naval	3
1.2.2	Autonomous Underwater Vehicles	5
1.2.3	Stakeholders	7
1.3	Problem Statement	7
1.4	Purpose	8
1.5	Scope and Delimitations	8
1.6	Report Structure	8
2	Methodology	10
2.1	Research Strategy	10
2.1.1	Action Research-Inspired Approach	10
2.2	Research Framework	13
2.2.1	Scientific Reasoning and Research Logic	13
2.2.2	Role of Theory and Empirical Data	14
2.3	Research Methods	15
2.3.1	Literature Review	15
2.3.2	Stakeholder Interviews	17
2.3.3	Benchmarking of Existing Solutions	19
2.3.4	Requirements Engineering	19
2.3.5	Tool Development Process	20
2.4	Method Reliability Considerations	20
2.5	Discussion of Alternative Methods	22
3	Requirements Elicitation	25

3.1	Literature Review	25
3.1.1	Life-Cycle Management	25
3.1.2	Integrated Logistics Support	27
3.1.3	Reliability, Availability, and Maintainability	31
3.1.4	RAM Prediction and Statistical Estimation	35
3.1.5	Reliability Growth and Trend Analysis	38
3.1.6	Non-Parametric Approaches for RAM	40
3.1.7	Reliability and Safety Analysis Methods	41
3.1.8	S5000F - International Specification for In-Service Data Feedback	46
3.2	Stakeholder Interviews	53
3.2.1	Stakeholder Roles and Interview Participants	53
3.2.2	Use Context and Applications	54
3.2.3	Analytical Needs and Outputs	54
3.2.4	Input Requirements and Data Structure	56
3.2.5	Adoption and Acceptance Criteria	57
3.2.6	Technical Constraints and Design Considerations	58
3.3	Benchmarking	60
4	Requirements Engineering	62
4.1	Product Requirements Document	62
4.1.1	Functional Requirements	63
4.1.2	Non-Functional Requirements	70
5	Development Process	71
5.1	Development Strategy	71
5.2	System Architecture	72
5.2.1	Front-End Architecture	72
5.2.2	Back-End Architecture	73
5.2.3	Database Design Considerations	74
5.3	Handling of Structured RAM Data	75
5.4	Implementation of S5000F-Compliant Data Export	75
5.4.1	Tailoring Strategy	76
5.4.2	RAM Metrics Export Messages	77

5.4.3	Action Report Export Messages	77
5.5	Iterative Development and Requirement Baselines	78
5.6	High-Level Design Documentation	79
6	The Developed System	80
6.1	The Digital Tool	80
6.1.1	Landing Page.....	80
6.1.2	Problem Reports	81
6.1.3	Analysis	82
6.1.4	Action Reports.....	90
6.1.5	Configurations	90
6.1.6	Users	92
6.2	The Excel Template	93
7	Discussion	98
7.1	Trade-Offs and Design Decisions	98
7.1.1	Deriving Measures from Feasible Operational Data ...	98
7.1.2	External Data Collection and Secure Import	99
7.1.3	Failure Definition and Inclusion in Analysis	100
7.1.4	Availability Metric: MTTR versus MDT	101
7.1.5	Weibull Modelling Choices	101
7.2	Limitations	103
7.3	Research Contribution.....	105
7.4	Future Work	106
7.4.1	Pilot Implementation and Evaluation	107
7.4.2	Scaling and Further Development	108
8	Conclusion	111
	Bibliography	113
	Appendices	120
	Appendix A: Interview Guide	120
	Appendix B: Notations	122
	Appendix C: Interviewees	123
	Appendix D: Product Requirements Document.....	124

List of Tables

2.1	Keyword clusters and example search terms used in FINN.	16
3.1	All S-Series suite specifications	47
3.2	All S-Series suite support specifications	48

List of Figures

1.1	Saab AB business areas with Naval highlighted.....	3
1.2	Example of an AUV62-AT. Adapted from [16].	6
2.1	The Shewhart Cycle.....	11
2.2	Overview of the iterative research process.....	13
3.1	Bathtub curve illustrating a typical early-life, useful-life, and wear-out failure behaviour.	33
3.2	Weibull probability plot based on mock time-to-failure data. The plotted points represent observed failures, while the fitted line shows the linearised Weibull relationship used to estimate the shape parameter β and the characteristic life η . The coefficient of determination, R^2 , indicates the goodness of fit...	37
3.3	Duane plot based on mock failure data for a repairable system. The plotted points represent the relationship between cumulative operating time and cumulative MTBF in logarithmic form, while the fitted line shows the linearised Duane model used to estimate the growth parameter α . R^2 indicates the goodness of fit.	39
3.4	Illustration of the FRACAS closed-loop process.	43
3.5	Schematic overview of the S-Series specifications. Adapted from [9].	49
3.6	Overview of the role of S5000F between actors. Adapted from [9].	50
6.1	Landing page of the RAM tool, where the user selects a product to work with.	81

6.2	Problem report list view with search and filter controls, and functionality for importing new problem reports into the dataset of the selected product.	82
6.3	Failure Distributions view summarising the filtered problem report dataset using distribution charts and a report-volume trend over time.	83
6.4	Distribution charts in the Failure Distributions view, summarising the filtered problem reports by key classifications (NFF, root cause, discipline, and failure area).	83
6.5	Bar chart of problem report volume in the Failure Distributions view, showing the number of reports in the filtered dataset aggregated by month or week.	84
6.6	Mutual filter component for applicable analysis views, used to define the subset of problem reports included in the analysis. ..	84
6.7	RAM Metrics view showing calculated metrics per selected segment(s), together with predicted and targeted values, and a cumulative metrics trend over time.	85
6.8	RAM Metrics view showing calculated RAM metrics for the selected segment(s), with comparisons to predicted and targeted values.	85
6.9	Plot of cumulative RAM metrics over time in the RAM Metrics view, illustrating how selected metrics evolve as additional operational data are accumulated.	86
6.10	Reliability Trends view providing Duane- and Weibull-based analysis for selected segment(s).	87
6.11	Duane reliability growth plot in the Reliability Trends view, illustrating reliability growth over accumulated operating time for the selected segment(s), together with the estimated growth rate (α) and the trend direction (positive/negative) with goodness-of-fit (R^2).	88

6.12	Weibull probability plot in the Reliability Trends view, showing the fitted two-parameter Weibull model for the selected segment(s) and the corresponding estimated parameters (β and η) with goodness-of-fit (R^2).	89
6.13	Action Report list view used to document follow-up actions and decisions, including corrective actions and change references.	90
6.14	Product configuration view used to maintain product-specific reference data such as units, breakdown structure, and RAM metric targets.	91
6.15	Failure Area configuration view used to maintain Discipline-specific Failure Area classification structure for structured reporting and filtering.	92
6.16	User management view for administering users and assigning role-based access within the tool.	93
6.17	Excel template - Problem Report Form (Variant 1). Overview of the initial reporting sheet used for problem reporting, with NFF.	94
6.18	Excel template - Problem Report Form (Variant 2). Overview of the initial reporting sheet used for problem reporting, without NFF and with Operator Error.	95
6.19	Excel template - Problem Report Form (Variant 3). Overview of the initial reporting sheet used for problem reporting, without NFF and with Component Error.	96
6.20	Excel template - Revision History. Complementary sheet used for revision history.	97
6.21	Excel template - Reference Data. Supporting sheet used to manage and add reference data for the template, such as values used in drop-down menus.	97

TERMINOLOGY

2F – 2 Frameworks

3F – 3 Frameworks

4F – 4 Frameworks

API – Application Programming Interface

ASW – Anti-Submarine Warfare

AUV – Autonomous Underwater Vehicles

CDM – Common Data Model

DSR – Design Science Research

FHA – Functional Hazard Analysis

FMECA – Failure Mode, Effects, and Criticality Analysis

FRACAS – Failure Reporting, Analysis, and Corrective Action System

FTA – Fault Tree Analysis

HLDD – High-Level Design Document

ILS – Integrated Logistics Support

IPS – Integrated Product Support

LCC – Life-Cycle Cost

MDBF – Mean Distance Between Failures

MDT – Mean Down Time

MLE – Maximum Likelihood Estimation

MTBF – Mean Time Between Failures

MTTF – Mean Time To Failure

MTTR – Mean Time To Repair

NFF – No Faults Found

NHPP – Non-Homogeneous Poisson Process

PFD – Probability of Failure on Demand

PHM – Prognostics and Health Management

PRD – Product Requirements Document

PSM – Product Support Management

R&M – Reliability & Maintainability

RAM – Reliability, Availability, and Maintainability

RAM-C – Reliability, Availability, Maintainability, and Cost

RAMCT – Reliability, Availability, Maintainability, Capability, and Testability

RAMS – Reliability, Availability, Maintainability, and Safety

RCA – Root Cause Analysis

RUL – Remaining Useful Life

SE – Sustaining Engineering

UML – Unified Modeling Language

UoF – Unit of Functionality

XML – eXtensible Markup Language

XSD – XML Schema Definition

1 Introduction

This chapter introduces the thesis by providing background information, including an overview of Saab Naval and Autonomous Underwater Vehicles (AUVs). It then presents the problem statement, outlines the purpose and scope, and concludes with a brief description of the report structure.

1.1 Background

Engineering systems in sectors such as defence and other capital-intensive industries are characterised by high complexity, high capital cost, and strong dependence on effective maintenance and support [1]. In this context, reliability, availability, and maintainability (RAM) are widely recognised as key system attributes, as insufficient RAM performance can lead to increased downtime, higher maintenance burden, and significant life-cycle cost impacts [1], [2]. Consequently, RAM engineering and associated methods of analysis are used to support life-cycle decision-making and systematic improvement efforts based on operational feedback, particularly in settings where RAM is critical [2], [3].

In maritime defence applications, RAM is closely tied to operational availability, where the overarching objective is to keep systems available for use when needed while controlling operations and support costs throughout the life cycle [4]. Guidance for naval acquisition and sustainment (maintenance and support) therefore emphasises that RAM factors should be addressed early and translated into actionable requirements and design activities, since insufficient RAM performance increases the risk of higher costs of ownership and operation and reduced availability [4], [5]. This also implies that RAM considerations are not limited to technical analysis in isolation, but must be integrated with systems engineering and product support decisions throughout the life cycle [4].

To enable such improvements in practice, structured operational feedback processes can be used to capture, analyse, and translate failure and maintenance

events into corrective actions [6]. A Failure Reporting, Analysis and Corrective Action System (FRACAS) is an example of a disciplined closed-loop approach intended to provide management visibility and control, and to use failure and maintenance data to implement corrective actions that prevent recurrence and simplify or reduce maintenance tasks [6]. In parallel, improvements in system reliability depend on identified issues being addressed through timely and effective corrections, while progress is tracked through periodic assessments against planned reliability targets, linking data collection and analysis directly to design and process improvements [4], [7].

A key prerequisite for effective RAM work is therefore the ability to consistently capture, structure, and reuse operational and maintenance feedback in a way that supports analysis across stakeholders and life-cycle phases. One approach to achieve this is to rely on established Integrated Product Support (IPS) specification frameworks, such as the S-Series suite, which is designed to provide aligned processes and a common data model for organising and exchanging product support information [8]. Within this suite, S5000F specifically structures in-service data feedback to handle operational, maintenance, and other stakeholder inputs and outputs, supported by the S-Series common data model to enable consistent information exchange and improved integration across the product-support information chain [8], [9].

In this context, Saab Naval is seeking to strengthen how RAM data are collected, structured, and utilised for AUVs to support long-term follow-up of RAM in operation. The challenge is not only to perform RAM analyses, but to establish a workable and consistent way of capturing operational and maintenance feedback so that it can be reused over time, traced to decisions and corrective actions, and presented in a form that supports evaluation and improvement. This motivates the development of a digital RAM tool tailored to AUVs, grounded in a structured in-service data feedback approach such as S5000F, with emphasis on how relevant data should be collected, organised, and visualised to improve reliability, extend product lifetime, and reduce life-cycle costs.

1.2 Organisational Context

Saab AB is a Swedish defence and security company organised into four business areas, as illustrated in Figure 1.1 [10]. This thesis is conducted in collaboration with Saab Naval, the business area focused on naval shipbuilding and underwater systems [10]. The organisational context is presented here to clarify where in Saab the study is situated and to frame the industrial setting in which the digital RAM tool is developed.

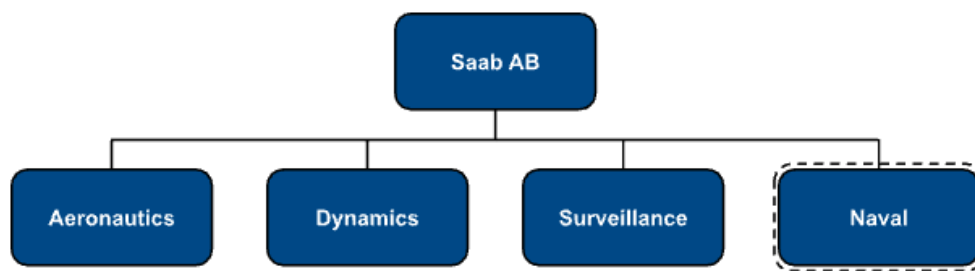


Figure 1.1: Saab AB business areas with Naval highlighted.

1.2.1 Saab Naval

Saab Naval is a naval shipbuilding and underwater systems organisation [10]. Saab Naval was until recently known as Saab Kockums and the company's roots trace back to the industrial operations established in Malmö in the 19th century, with Kockums developing from mechanical workshop activities into shipbuilding and, over time, becoming closely associated with the construction of naval vessels and submarines [11]. In its current form, Saab Naval was established when Saab acquired the Swedish shipyard TKMS AB (formerly Kockums) from ThyssenKrupp, an acquisition that took place in 2014, thereby integrating Kockums into Saab's broader defence and security portfolio [12]. Today, Saab Naval is headquartered in Lund, recently relocated from Malmö. It has operations connected to multiple Swedish naval sites, supporting development, production, and long-term in-operation support of naval and underwater systems [10], [12].

Within Saab's organisational structure, Naval is described as a business area

focused on the development, production, and in-service support of submarines, surface vessels, and naval systems [10]. The portfolio also includes underwater systems such as remotely operated and autonomous vehicles, as well as related subsystems and components for integration in naval solutions for customers in Sweden and internationally [10].

Characteristically for Saab Naval, their responsibility extends beyond initial design and delivery. As mentioned, Naval's scope includes development, production, and in-service support, i.e. activities that span multiple life-cycle phases and require long-term technical follow-up. Depending on contractual arrangements and customer capabilities, maintenance and repair activities may be performed either by the customer organisation, by Saab Naval, or in a shared set-up. The detailed allocation of responsibilities, processes, and the extent of customer versus supplier repair is customer- and contract-specific and is therefore treated as confidential in this thesis. Consequently, the work addresses RAM data handling in a way that is applicable regardless of whether corrective actions are carried out by the operator, the supplier, or both. In practice, a life-cycle responsibility also implies a need for systematic performance follow-up after delivery. For long-lived products, achieving and sustaining required capability is closely linked to how RAM is managed over time, i.e. how failures, downtime, maintenance effort, and corrective actions are tracked and translated into improvements. This typically requires a structured feedback loop, where operational and maintenance feedback is captured and analysed, corrective actions are implemented, and the effects are monitored over time to enhance RAM.

At present, however, Saab Naval does not have a unified, structured approach for collecting and utilising such RAM-related feedback over time for their AUVs. This limits the ability to perform consistent follow-up, maintain traceability from observed events to corrective actions, and reuse information across life-cycle phases. In addition, the data environment is characterised by strict requirements related to security classification, export control, and customer-specific constraints, which influences how operational information can be stored, accessed, and shared. This further increases the need for a

controlled and well-structured solution. The work in this thesis therefore aims to contribute towards establishing such a feedback capability by developing a digital RAM tool adapted to AUVs, designed to support systematic data capture, structuring, analysis, and presentation of RAM information for long-term follow-up and improvement.

1.2.2 Autonomous Underwater Vehicles

Saab Naval's offering is divided into several business units, such as Submarines, Combat Systems, and Underwater Systems [13]. Within Underwater Systems, Saab Naval presents a portfolio of products intended for a range of operational needs [13]. This thesis is limited to the AUV segment within Underwater Systems. More specifically, the thesis focuses on the the AUV62 family, which represents one of Saab Naval's fully autonomous solutions. In this product context, Saab Naval's customers are typically defence organisations and other governmental stakeholders operating the systems in national service.

Within the AUV62 family, two configurations are considered: AUV62-AT and AUV62-MR. AUV62-AT (Acoustic Target, most commonly) is presented as an advanced at-sea anti-submarine warfare (ASW) training system, used as a programmable target to train surface, air, and subsurface ASW assets [14]. The vehicle uses a transducer tail to mimic submarine acoustic profiles and the system supports post-exercise analysis for debriefing and training development [14]. In contrast, AUV62-MR (Mine Reconnaissance) is designed for mine countermeasures and mine reconnaissance missions [15]. It autonomously surveys large areas and generates high-resolution sonar data to support detection and classification of objects on the seabed [15]. An example of the AUV62 vehicle platform, more specifically an AUV62-AT, is shown in Figure 1.2.

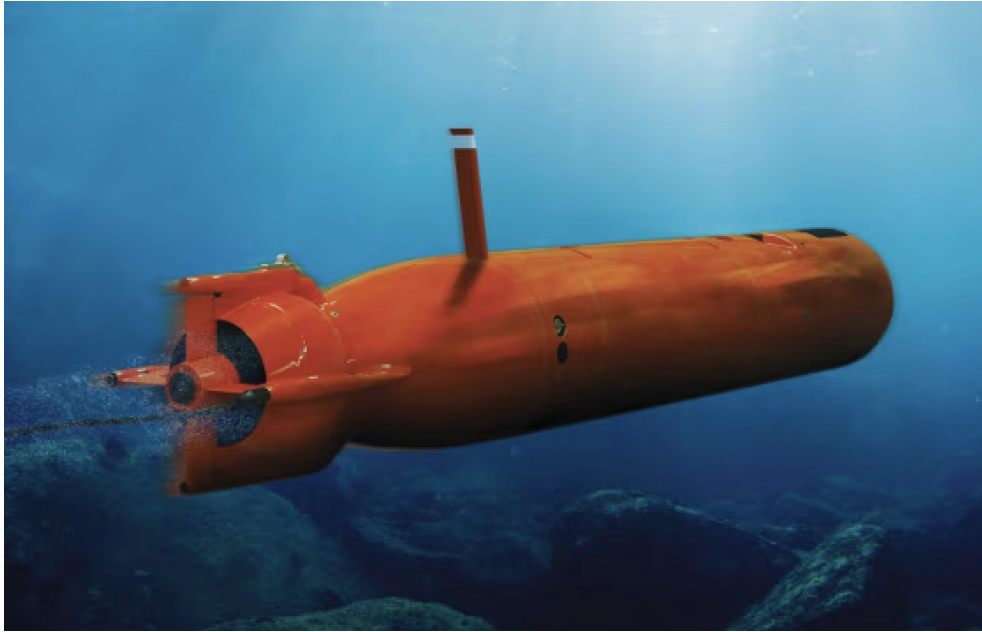


Figure 1.2: Example of an AUV62-AT. Adapted from [16].

AUV62 operations are generally described as mission-based, supported by dedicated planning and evaluation functions [14], [15]. For example, AUV62-AT uses a mission planning system to create exercise scenarios and supports post-exercise analysis for debriefing, while AUV62-MR uses mission planning and post-mission analysis together with a launch and recovery system for deployment and recovery [14], [15]. From a life-cycle perspective, this places emphasis on systematic follow-up of operational outcomes and maintenance actions over time. Even without detailed technical modelling, it is useful to recognise that mission profiles, operating conditions, and support arrangements affect what types of failures occur, how downtime is generated, and what maintenance effort is required. Consequently, a structured way of capturing operational and maintenance feedback becomes important to enable consistent follow-up of RAM, support corrective actions, and provide input to long-term planning such as upgrades, spares strategies, and life-time extension initiatives.

1.2.3 Stakeholders

The digital RAM tool developed in this thesis is intended to support multiple stakeholder groups across the product life cycle. Internally at Saab Naval, this includes roles involved in RAM/ILS work, systems engineering, and maintenance planning, as well as functions responsible for quality follow-up and continuous improvement, including management. Externally, the primary stakeholders are the customer organisations operating the AUVs and performing parts of the maintenance and repair work, as well as personnel involved in operational planning, reporting of failures and maintenance actions, and long-term sustainment planning. Since responsibilities may be shared between operator and supplier depending on contractual arrangements, the tool is positioned to be beneficial for both internal and external stakeholders through structured feedback.

1.3 Problem Statement

Saab Naval wishes to strengthen how RAM data are collected, structured, and used for its AUVs, in order to improve RAM over time, extend product lifetime, and reduce life-cycle costs. Today, some RAM related data may be available, but a structured method for its systematic collection, use, and analysis over time is non-existent. Consequently, the absence of a structured method also means that there is currently no coherent digital RAM tool adapted to AUVs, limiting consistent follow-up, traceability, and long-term utilisation of RAM/FRACAS information.

In addition, the underlying data are often highly security-classified, export-controlled, and customer-sensitive. This restricts the suitability of many commercial RAM tools, especially cloud-based solutions or systems relying on data storage outside Sweden. This motivates an in-house solution where access control, data handling, and storage can remain within Saab's secure IT environment. The central problem is therefore to identify how established RAM methodology and relevant standards, particularly S5000F for system interoperable RAM data exchange, can be translated into a practical and

implementable digital tool tailored to AUVs.

1.4 Purpose

The purpose of the thesis is to develop a digital RAM tool tailored for AUVs. The tool will be based on an analysis of how in-service data feedback can be collected, structured, and presented to support evaluation of RAM.

1.5 Scope and Delimitations

This thesis is limited to the development of a digital RAM tool concept and prototype tailored to Saab Naval's AUVs. The work focuses on the AUV62 family, specifically the AUV62-AT and AUV62-MR configurations, and the RAM-related operational and maintenance feedback associated with these vehicles.

From a standards perspective, the study uses S5000F as the primary reference for structuring and handling exchange of in-service data feedback. The scope does not include the full S-Series suite, and the outcome should be viewed as an interpretation and application of relevant S5000F principles rather than a complete implementation of all specification requirements. Consequently, the developed data structures and workflows aim to be aligned with S5000F where feasible, but full compliance verification is outside the scope of this thesis.

Finally, the work is limited to designing and developing the digital tool (including its data input, handling, and presentation concepts) and demonstrating it as a prototype. Deployment, integration into Saab Naval's IT environment, organisational implementation, and long-term operation and maintenance of the tool are not included. The prototype is instead intended to be handed over as a basis for further development and implementation within Saab Naval.

1.6 Report Structure

The report is structured as follows:

Chapter 1 introduces the thesis and its context. It provides background on RAM and presents Saab Naval and its AUVs. The chapter then defines the problem addressed in the thesis by outlining its purpose and scope.

Chapter 2 outlines the methodology used in the project, including the approach for the literature review, requirements elicitation, and the subsequent analysis.

Chapter 3 details the requirement elicitation, including literature review, covering topics such as Life-Cycle Cost (LCC) and Integrated Logistics Support (ILS), RAM, reliability growth, FRACAS, and the S5000F standard, as well as insights from the conducted internal stakeholder interviews.

Chapter 4 provides the requirements engineering, i.e. the analysis and specification, including the functional and non-functional requirements identified for the digital tool.

Chapter 5 describes the development process of the digital tool, including its architecture and design decisions.

Chapter 6 presents the developed digital RAM tool and the Excel-based reporting template, describing how they support structured data capture, import, analysis, follow-up documentation, and configuration management.

Chapter 7 discusses and evaluates the results in relation to the project scope, including implications, limitations, and directions for future work.

Chapter 8 concludes the thesis by summarising the main findings and addressing the problem statement.

2 Methodology

This chapter outlines the methodology adopted in this thesis. It includes descriptions of the research strategy and framework, followed by the concluded research methods with reliability considerations. Finally, a discussion of alternative research methods is presented.

2.1 Research Strategy

This section describes the overall research strategy adopted in this thesis. It explains how the study was structured to address the problem and achieve the stated objectives. Given the applied and industrial nature of the thesis, the chosen strategy emphasised close collaboration with stakeholders, iterative development, and continuous reflection throughout the research process.

Conceptually, this thesis can be described as a problem-solving study fuelled by insights from an exploratory study. Problem-solving studies aim to develop solutions or improvements for an identified problem area [17]. On the other hand, exploratory studies seek to build an in-depth understanding of how a phenomenon or process works [17]. The content presented in the subsequent chapters was conducted primarily using an Action Research-inspired approach. Action Research involves a carefully planned, closely monitored, and systematically documented investigation carried out in collaboration with the studied context, with the purpose of solving a problem or improving a process [17].

2.1.1 Action Research-Inspired Approach

The research strategy adopted in this thesis is inspired by Action Research, due to its suitability for applied, organisation-related problem solving and its emphasis on collaboration with practitioners [18]. In its full form, Action Research entails an iterative cycle where a problem is identified, a change is

proposed and carried out in practice, and the outcome is evaluated and reflected upon to guide further action [17]. This logic is closely related to continuous improvement cycles, such as the Shewhart cycle (Plan–Do–Study–Learn), illustrated in Figure 2.1, where evaluation informs subsequent iterations [17].



Figure 2.1: The Shewhart Cycle.

However, it is important to note that this thesis does not follow Action Research in a strict sense. While the study fully covers the initial phases of identifying and analysing an organisational problem and proposing a solution, the scope does not include full-scale implementation of the complete tool in operations (e.g. organisational rollout, establishment of routines, and long-term use). As discussed by Höst et al., technical thesis projects often result in a prototype or partial solution that can be evaluated and refined, whereas making the solution permanently usable in an organisation typically requires additional work related to robustness, usability, integration, and documentation that lies beyond the scope of an academic thesis [17].

Accordingly, the iterative elements in this work are realised primarily through repeated cycles of requirements elicitation, design, and stakeholder feedback. Instead of evaluating organisational change through sustained operational deployment, evaluation and learning were achieved by iteratively validating requirements and design decisions with relevant stakeholders and refining the

tool concept and development based on their feedback. In this way, the study applies a pragmatic, Action Research-inspired working process, participatory and iterative, while acknowledging the limitations in terms of organisational intervention and implementation scope.

As the study was conducted in an Action Research-inspired manner, the thesis workers were actively involved in both the investigation of the problem and the design and development of the proposed solution. Critically evaluating solutions to which one has contributed introduces a risk of bias [17]. To mitigate this risk, external perspectives were utilised through recurring validation with relevant stakeholders, both to challenge the thesis workers' interpretations and to cross-check stakeholder requirements and assumptions.

Involving internal stakeholders as active contributors provides access to insider knowledge of the problem area [18]. At the same time, this involvement may introduce the risk that participants treat certain important factors as self-evident or trivial and therefore do not articulate them explicitly [18]. To address this, stakeholder input was complemented with an extensive literature review to validate and refine empirical insights against established theory, standards, and documented best practices. This iterative combination of empirical findings and theory aligns with the logic of an Action Research-inspired working process, even though full organisational implementation was outside the thesis scope.

Initial requirements from internal stakeholders were used to delimit and guide the literature review, ensuring that subsequent semi-structured interviews were informed by sufficient domain knowledge. The resulting empirical and theoretical insights were then synthesised into a requirements specification, combining theory-informed principles and standard-aligned practices with the practical constraints and expectations expressed by stakeholders.

Based on the synthesised requirements, the digital RAM tool was developed iteratively and progressively. Continuous delivery of intermediate versions enabled recurring feedback from stakeholders, which supported ongoing validation of both requirements and design decisions. Although this evaluation was primarily conducted through iterative stakeholder review rather than full-

scale deployment in operations, the development process follows the iterative logic reflected in the Shewhart Cycle. Figure 2.2 provides an overview of the research process and the interaction between requirements elicitation, requirements engineering, and tool development.

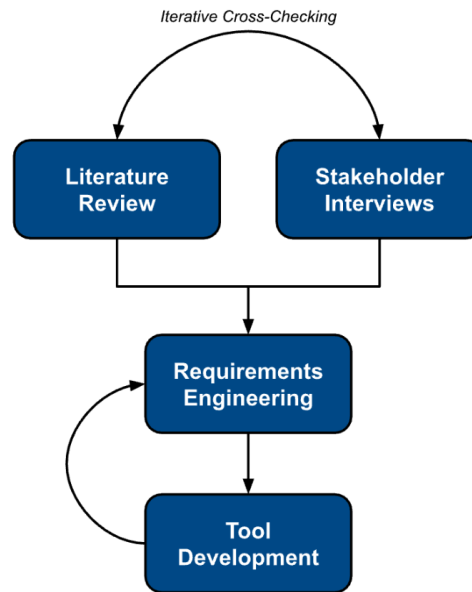


Figure 2.2: Overview of the iterative research process.

2.2 Research Framework

This section describes the conceptual framework that establishes how insight is derived within the study. While the research strategy defines the overall structure and context of the work, the research framework clarifies how theoretical perspectives and empirical observations are combined to inform analysis, decision-making, and design choices throughout the thesis.

2.2.1 Scientific Reasoning and Research Logic

This thesis adopted an abductive research logic, i.e. iteratively linking observations and theory, to guide how insights were derived throughout the work. Empirical findings from stakeholder interviews were continuously examined

in relation to established theory and standards. These findings included expressed user needs, descriptions of current FRACAS/RAM-related workflows, constraints related to IT environments, and prioritised functional and non-functional requirements for the tool. In parallel, theoretical conclusions were compared against the observed practical constraints and requirements. This simultaneous collection of data and theory building resonates well with abductive reasoning and Action Research [19]. It enabled the research to progressively shape the understanding of the problem and the developed solution.

Alternatives to abductive reasoning are deductive and inductive reasoning [19]. In deductive reasoning, conclusions are derived from an existing theoretical framework through the formulation and testing of predefined hypotheses or propositions [19]. In inductive reasoning, the aim is to create generalised theoretical conclusions based on empirical findings, often without the need to rely on theoretically established knowledge [19].

The research conducted in this thesis did not follow deductive reasoning, as it did not aim to test predefined hypotheses. Nor can it be fully regarded as inductive, since it utilised existing theory and the prime objective was not to produce generalised conclusions. In addition, the latter argument is a limitation commonly associated with Action Research, as insights are derived from and closely tied to a specific organisational context and problem area, which inherently constrains the generalisability of the results [18].

2.2.2 Role of Theory and Empirical Data

The research was grounded in a combination of empirical input from stakeholders and established theory within the RAM area. Empirical data, primarily gathered through stakeholder interviews, provided insight into practical needs, constraints, and expectations related to the use of RAM data in an industrial context. At the same time, theory and standards identified through the literature review provided a structured and generalised perspective on RAM methodology, data handling, and best practices.

Rather than treating either empirical findings or theoretical perspectives in

isolation, the study relied on an iterative interplay between the two. Empirical observations were continuously examined in relation to established theory and standards, while theoretical assumptions were balanced against stakeholder-identified practical constraints. This relationship served as a form of analytical cross-checking, helping to identify potential biases, challenge assumptions, and maintain analytical distance despite the thesis workers' active involvement in the development work.

This approach also supported handling differing perspectives between stakeholders' expectations and theory-informed design considerations. Such differences were not viewed as contradictions to be resolved in favour of one perspective, but as opportunities for reflection and refinement of requirements and design decisions. By combining empirical observations with established theory, the resulting requirements and tool design aimed to achieve both practical applicability and methodological correctness.

2.3 Research Methods

This section describes the research methods applied in the thesis and how they were used to collect and analyse data in order to address the problem. The methods are presented in the order in which they were primarily applied, while still having an iterative approach in mind.

2.3.1 Literature Review

The purpose of the literature review was to build a well-founded understanding of the problem area by identifying and reviewing relevant, credible sources. It provided the theoretical and methodological basis for the work by clarifying key concepts, established approaches, and previous findings. The literature review was treated as an iterative activity, where insights gained during the project led to refined search terms, additional sources, and updated interpretations, ensuring that the review continuously supported the subsequent phases of the thesis [17].

The literature review was conducted using the iterative three-phase search approach starting with an initial broad scan of the field, moving to a focused screening of relevance and credibility, and finalised with a more targeted follow-up search based on the most central sources [17]. Searches were primarily performed through FINN (Lund University Library search service), which was used as the main entry point to access journal articles, books, and other academic material.

Initially, the search strategy was deliberately kept wide in scope in order to capture the terminology used across both reliability engineering and FRACAS-related research. Keywords were organised into thematic groups reflecting the purpose of the thesis, RAM, FRACAS and failure reporting processes, and work related to requirements and tool development. These clusters were combined using the advanced search function in FINN and adjusted as the search progressed, for example by adding alternative terms and abbreviations encountered in the initial results. A summary of the keyword clusters and example search strings are presented in Table 2.1.

Table 2.1: Keyword clusters and example search terms used in FINN.

Thematic group	Example keywords / search terms
RAM & reliability concepts	RAM; reliability; maintainability; availability; reliability engineering; reliability analysis; maintenance; maintenance planning; maintenance data
FRACAS & failure reporting processes	FRACAS; “failure reporting”; “failure analysis”; “corrective action”; “root cause analysis”; “in-service data”; “operational data”; “fault reporting”; “incident reporting”; “nonconformance”
Requirements & tool development	requirements; “requirements elicitation”; “requirements specification”; “system requirements”; “user needs”; “stakeholder needs”; “tool development”; “information system”; “data model”; “decision support”; “workflow”

After the initial search, the returned sources were screened to identify a manageable and relevant set for closer reading. This screening was based on titles and abstracts and aimed to exclude material outside the thesis scope, while retaining sources that directly addressed relevant information. In parallel, the credibility of candidate sources was considered, for instance by prioritising peer-reviewed publications and widely recognised standards or handbooks where appropriate.

The final phase used the selected core literature to deepen the review and reduce the risk of missing relevant work. The most relevant sources were read in greater detail to identify key concepts and recurring topics, which also provided the input for more precise searches in FINN. In addition, reference lists in the core sources were used to locate earlier foundational publications (backward searching), and when available through FINN's linked databases, newer publications connected to the same core sources were also used to broaden coverage.

In addition to the above-mentioned search process, the S5000F specification was treated as a central source in the literature review. Since S5000F defines terminology, processes, and information requirements related to in-service data feedback, it was used for understanding what data should be captured and how it should be structured. The work with S5000F consisted of repeatedly reviewing the relevant sections and extracting statements that could be translated into tool-relevant requirements (e.g. required data elements and data exchange flows). These extracted points were later cross-checked against academic literature and industry requirements to clarify interpretations, identify practical considerations, and ensure that the resulting requirements were both standard-aligned and feasible in the company context.

2.3.2 Stakeholder Interviews

To gather relevant input from the different stakeholders of the digital RAM tool, interviews were conducted with key representatives within the company in parallel with the literature review. Interviews were selected as the primary data

collection method in order to capture stakeholders' perspectives, requirements, and preferences regarding the tool.

The study mainly employed qualitative, flexible interview formats, specifically semi-structured and unstructured interviews. In semi-structured interviews, the interviewer uses a prepared interview guide with a set of themes and questions as a starting point [20]. However, the questions do not need to be asked in a fixed order, and the respondent is encouraged to elaborate and reflect beyond direct answers [20]. Follow-up questions can also be introduced during the conversation, even if they were not part of the original guide [20]. Compared to a structured, more quantitatively oriented interview, the format is therefore more open, while still ensuring that certain topics are covered [20].

Unstructured interviews, in contrast, typically begin from only a small number of discussion points [20]. The respondent is largely free to steer the conversation and develop their reasoning in their own terms, making the interaction closer to an ordinary conversation between the interviewer and the participant [20].

Building on these interview approaches, the requirements elicitation was organised around semi-structured interviews complemented by ongoing dialogue (unstructured interviews) with relevant stakeholders at Saab Naval. The overall objective was to translate stakeholder knowledge into actionable requirements by identifying both user needs and practical constraints related to existing ways of working and the current IT environment.

Semi-structured interviews were conducted with two main stakeholder groups. Firstly, interviews were held with internal RAM/ILS experts (including the thesis supervisors), who also represent potential output users of the system. Secondly, semi-structured interviews were carried out with stakeholders representing the input side of the workflow, i.e. operational roles responsible for capturing and submitting the data required by the system, such as ILS engineers and related functions. Together, these groups provided complementary perspectives on requirements, covering both analytical use cases and the practical realities of data capture in day-to-day operations. The interview guide used for the semi-structured interviews is provided in Appendix A.

In addition to the semi-structured interviews, a continuously ongoing dialogue was maintained with the supervisors and IT personnel throughout the project. These informal discussions focused on the tool's intended functionality, properties, and overall design. These conversations were used to iteratively clarify assumptions, discuss relevant theory, obtain assistance with IT-related questions, and refine requirements as the understanding of the problem and the solution evolved.

2.3.3 Benchmarking of Existing Solutions

A limited benchmarking activity of an existing commercial RAM software solution was conducted. The purpose was not to extend the synthesised requirements, but to provide an external reference point for discussions of the developed tool and requirements analysis.

Through a scheduled demonstration session of a vendor-provided software solution, commercially prioritised functional and design choices could be observed. Due to limited access and the demonstrative nature of the activity, the benchmarking focused on qualitative observations rather than detailed feature-by-feature comparison.

2.3.4 Requirements Engineering

Once preliminary insights had been derived from the literature review, the stakeholder interviews, and, to some extent, the benchmarking, the work to synthesise the product requirements was initiated. The produced requirements were based mainly on analysis of the findings from both literature and interviews. It was an iterative process, where the findings were continuously compared, scrutinised, and aligned. Through this approach, different perspectives and priorities were never considered in isolation.

The synthesised requirements were collected and structured in a Product Requirements Document (PRD), to support guidance and traceability throughout the development process. The PRD was also a tool to initially validate the product's usability, ensuring captured functional and non-functional require-

ments were aligned with stakeholder expectations. It was a living configuration item, which provided a well-motivated foundation for guiding the iterative development of the digital RAM tool.

2.3.5 Tool Development Process

In parallel with the requirements synthesis, the product system design was determined. The major factors in shaping the system design were insights regarding Saab Naval's secure IT environment and requirements of the S5000F specification. The IT environment implied specific constraints, possibilities, and guidelines related to, among other things, infrastructure and deployment. The S5000F specification emphasised well-defined information entities and traceability between data elements.

The tool was developed using an iterative and incremental process that ran in parallel with continued requirements engineering. As requirements were refined, the development was adapted. Versions of the tool were demonstrated to relevant stakeholders during the development process, obtaining feedback used as input for subsequent iterations, aligned with the overall research strategy.

During the development process, a High-Level Design Document (HLDD) was composed to present main architectural decisions and interactions between key components. Through the HLDD, the concluded design choices were stated in a traceable manner, thereby supporting the discussion of the finished product. In addition, the HLDD provided a basis for knowledge transfer during handover, for further development of the product.

2.4 Method Reliability Considerations

In this thesis, method reliability refers to the consistency and traceability of the research process, i.e. whether the applied methods were carried out in a structured manner such that similar procedures would likely lead to comparable findings.

For the stakeholder interviews, reliability was supported by using a semi-structured interview guide to ensure that the same main themes were covered across interviews (Appendix A), while still allowing follow-up questions when needed. To enable an accurate and consistent representation of stakeholder statements, the interviews were audio-recorded and transcribed, reducing the risk of misquotations and retrospective reinterpretation compared to relying solely on memory or fragmented notes. Interviews were, as far as practically possible, conducted under comparable conditions (similar time frames, an undisturbed setting, and the same interviewers), which supports consistency across sessions. Some interviews were conducted on-site and others remotely via Microsoft Teams. However, this was assessed as having limited influence on the results since the same interview guide, roles, and thematic focus were used in both settings, and the interaction still enabled explorative questions and clarification in real time. Immediately after each interview, interpretations were summarised and consolidated to reduce the risk of misinterpretation and to create a traceable basis for subsequent requirements work. In addition, the interpretations and synthesised requirements were repeatedly validated through follow-up dialogue with some of the stakeholders, which served both as an accuracy check and as a way to validate that the captured meaning was consistent with their intent.

For the literature review, reliability was supported through a structured and documented search and screening procedure. Searches were conducted using mainly the same database (FINN) and organised into predefined keyword clusters (Table 2.1), enabling repeatability of the search strategy. The selection of sources followed an explicit screening approach based on relevance (alignment with the thesis scope) and credibility (e.g. peer-reviewed research and recognised standards). The iterative search process (broad scan, focused screening, and follow-up searching) was applied in a structured manner, which supports transparency in how the final set of sources was established.

For the benchmarking activity, reliability was supported by treating the session as a structured observation. Since the benchmarking was limited in depth due to access constraints and the demonstration format, it was complemented

with a review of publicly available information on the vendor's website. This provided a consistent reference for clarifying terminology, stated capabilities, and positioning beyond what was shown in the presentation. The demonstration followed a predefined focus aligned with the synthesised requirements, i.e. observing commercially prioritised functional and design choices relevant to the thesis scope. Observations were documented directly during and immediately after the session to reduce recall bias. Since the benchmarking was limited in depth and mainly based on a vendor-led demonstration, its role was intentionally restricted to providing an external reference point rather than serving as a primary basis for requirements. This delimitation reduces the risk of over interpreting observations that may be dependent on the specific demo context. Despite these measures, the interview-based findings were influenced by practical constraints, such as the limited number of interviews, time restrictions, and the availability of stakeholders, which may have affected the width of perspectives captured. The study therefore emphasised iterative validation and traceability of decisions, making assumptions, trade-offs, and requirement reasoning explicit.

2.5 Discussion of Alternative Methods

While the methodology adopted in this thesis was selected to fit the applied and industry-driven nature of the work, several alternative research strategies and methods could also have been considered. The purpose of this section is to briefly discuss such alternatives and explain why they were not chosen. By contrasting these options with the selected Action Research-inspired strategy and abductive research logic, the section aims to clarify the trade-offs involved in method selection and strengthen the justification for the chosen approach. The discussion focuses on methodological fit with the thesis objectives, the need for close stakeholder involvement, and the iterative development of the requirements specification and the digital RAM tool within a real organisational context.

A survey-based quantitative approach, using questionnaires distributed within

each stakeholder group, could have been used as an alternative to the qualitative interview strategy. This approach can provide broader coverage and enable generalisation of attitudes, perceived needs, and priorities across a larger number of respondents, which may support statistical comparison between user groups [20]. However, survey methods typically offer less depth and limited opportunity for clarification and exploration of context-specific reasoning, compared to qualitative interviewing [20]. Given that this thesis aims to elicit detailed requirements and uncover practical constraints related to RAM/FRACAS workflows and data handling in a specific organisational setting, semi-structured interviews were considered more suitable for capturing nuanced stakeholder knowledge and translating it into actionable requirements.

A further alternative would have been to frame the work explicitly as a Design Science Research (DSR) study, where the primary outcome is a designed artefact (which in this case would be the digital RAM tool) and knowledge is produced through iterative build-and-evaluate cycles [21]. DSR typically emphasises that the artefact is developed and rigorously evaluated, and that the design process is grounded in an existing knowledge base so that the resulting contribution is both practically useful and methodologically justified [21]. While this thesis shares similarities with DSR through its artefact-oriented outcome and iterative development, DSR was not adopted as the main research strategy. The chosen strategy was considered a better fit for the continuous stakeholder involvement required to elicit, refine, and validate requirements in a real industrial context.

An additional alternative would have been to use an ethnographic approach based on participant observation, where the researchers spend time in the operational setting and document work practices, interactions, and information flows as they occur [20]. This method can provide rich, contextual insight into how relevant data are created and handled in practice, and can help reveal aspects that stakeholders may not mention explicitly in interviews [20]. However, participant observation typically requires substantial time in the field and sustained access to everyday work situations, which can be difficult to achieve within the time frame of the thesis. The industrial environment with

confidentiality and security constraints also creates additional difficulties in this particular case. For these reasons, stakeholder interviews combined with iterative validation were considered a more feasible way to elicit requirements while still capturing relevant practical perspectives.

In summary, the approaches discussed above offer distinct strengths, such as broader quantification of needs through surveys, artefact-focused development through DSR, and deep contextual understanding through participant observation. However, each alternative also involves trade-offs that reduce its suitability for this thesis, either by limiting depth and interaction, requiring more formalised development and evaluation cycles, or demanding extensive field access. There are also additional methodological alternatives beyond those presented here, but they involve similar compromises in relation to the project constraints and objectives. Given the applied objective for this thesis, the need for continuous stakeholder involvement, iterative refinement of both requirements and the digital tool within a specific organisational context, the selected Action Research-inspired strategy with an abductive research logic was assessed as the most appropriate methodological approach.

3 Requirements Elicitation

This chapter presents the process used to elicit the requirements that form the thesis' requirement specification. The requirements are derived from two complementary sources: a literature review establishing relevant theory, standards, and best practices, and stakeholder interviews capturing practical needs, workflows, and constraints in the target environment. Together, these inputs provide a traceable requirements foundation that informs the requirements engineering and guides subsequent design and development of the digital RAM tool.

3.1 Literature Review

This literature review establishes the theoretical foundation for the thesis and the subsequent requirements engineering. The notation used in this section is provided in Appendix B. It introduces the core concepts of RAM and outlines how operational and maintenance data can be structured to support analysis, learning, and decision-making over a system life cycle. The review covers established practices such as FRACAS and reliability growth. Finally, relevant standards within the S-Series suite, particularly S5000F, are framed as a basis for defining consistent information structures and workflows.

3.1.1 Life-Cycle Management

A product life cycle describes the phases a system passes through from initial concept and development, via production and operation/support, to retirement and disposal [22]. LCC captures the total cost of owning and operating the system across these phases, rather than focusing only on acquisition cost [22], [23]. This perspective is particularly relevant in maritime defence, where systems are operated and supported over long service lives and sustainment becomes a dominant driver of performance and affordability [4]. Since reliabil-

ity, availability, and maintainability directly influence downtime, maintenance workload, and support needs, RAM performance is closely tied to life-cycle outcomes and cost [4]. It is therefore essential to address RAM early and translate it into actionable requirements and design activities to avoid downstream availability shortfalls and increased LCC [4].

Life-Cycle Cost

In practice, LCC is commonly decomposed into major cost categories that span development, investment/acquisition, operating and support, and disposal [23]. Within defence cost analysis, operating and support cost is typically treated as a separate, major category and is structured using standard cost element breakdowns to improve consistency and traceability [24]. Such breakdowns enable analysts to distinguish, for example, between maintenance, supply/logistics-related elements, and personnel-related costs [24]. This could be useful both for cost estimation and for identifying which drivers can be influenced through design or support decisions.

A core challenge in LCC work is that results are sensitive to scope boundaries, assumptions, and data quality, meaning that credible LCC estimates require clearly stated ground rules and a transparent documentation of what is included and excluded [25]. It is therefore emphasised that high-quality cost estimates should be comprehensive, well-documented, and based on validated assumptions, since omissions or inconsistent definitions reduce comparability and can lead to misleading conclusions [25]. In long-lived systems, long time horizons introduce substantial uncertainty, which makes it particularly important to treat LCC as a decision-support tool used to compare alternatives under stated assumptions rather than as a single precise number [25].

From a RAM perspective, the practical relevance of LCC is that many dominant cost drivers are related to maintenance effort, part demand, and other sustainment activities that accumulate during operation and support [24]. Military defence guidance on reliability and maintainability engineering highlight that improving RAM can reduce maintenance burden and increase availability,

thereby influencing downstream sustainment outcomes, such as reduced downtime and maintenance effort, as well as total ownership cost [4]. Consequently, an LCC-informed approach motivates structured collection of operational and maintenance data, as these are required to quantify and manage sustainment cost drivers and to evaluate the impact of design or support changes over time [24]. These cost drivers are largely shaped by how the system is supported in practice, which motivates the ILS perspective as a structured way to plan and manage the resources, processes, and information needed to achieve required availability at acceptable LCC.

3.1.2 Integrated Logistics Support

This section introduces ILS as a life-cycle support framework, outlines its structural elements and evolution toward Integrated Product Support (IPS), and establishes its relevance for reliability engineering and in-service data within the context of this thesis.

ILS as a Life-Cycle Support Framework

ILS is an approach intended for achieving disciplined management of the activities needed to ensure that a product can be effectively and economically supported throughout its life cycle [26], [27]. It includes considerations from production and system acquisition to operation, establishing support capability with the intention of meeting performance objectives within acceptable LCC limits [26], [27]. Rather than functioning only as a downstream logistics activity, ILS is characterised by the integration of logistics considerations within system engineering activities, from the initial development phases to the operations and maintenance of the product [28]. Therefore, to be able to realise the benefits of ILS, it must be addressed proactively rather than reactively, ensuring that supportability is embedded in the design of the product. Supportability, in this setting, refers to the ability of a system to be supported and sustained effectively in its intended operational environment.

In practical terms, the fundamental objective of ILS is to achieve required

operability and readiness, i.e. the degree to which a system is ready and able to perform its intended mission when required, while minimising total LCC [27]. Although some literature distinguishes between buyer and supplier incentives for applying ILS, the underlying logistical principles converge toward shared objectives related to LCC and readiness assurance. To ensure that a delivered system remains supportable over its intended service life, logistical considerations, further depicted in the following subsection, must be systematically incorporated into planning, funding, and control processes [27]. This requires that individual logistics elements, which are the responsibility of logistics engineers, are coordinated and integrated within the system engineering of a product [27].

In an organisational context, effective implementation of ILS requires that logistics engineers operate as an integral part of the broader engineering effort [26]. They collaborate with other engineering disciplines during system development to ensure that supportability considerations are incorporated into the design process [26]. To determine the resources necessary to sustain the system in operation and to evaluate overall supportability, structured logistics analyses, such as RAM and LCC analyses, are conducted [26]. Furthermore, logistics support resources are identified as a major contributor to total LCC of military systems, which underscores the importance of systematic ILS application and organisational commitment to supportability throughout the system life cycle [26].

Based on the reviewed literature, and particularly the goal formulations presented by Jones [26], the primary objectives of ILS can be synthesised as:

- Minimising total LCC
- Achieving required operability and readiness levels
- Influencing design decisions to improve supportability
- Identifying and limiting cost drivers and support activities
- Defining and developing the resources required to support the system throughout its life cycle

This synthesis reflects the core engineering intent of ILS, where supportability

is embedded into system design in order to balance operational readiness requirements with LCC efficiency.

The Ten ILS Elements

ILS is organised around ten elements, as specified in DoD-D-5000.39 [29]. Eight of these elements primarily concern the logistics support resources required to achieve system operation readiness within its operational phase [30]. The remaining two elements, Maintenance Planning and Design Interface, differ in character, as they are directly connected to system engineering management activities [30].

The ten elements of ILS, as specified in DoD-D-5000.39 [29], are as follows:

- Maintenance Planning
- Manpower and Personnel
- Supply Support
- Support Equipment
- Technical Data
- Training and Training Support
- Computer Resources Support
- Facilities
- Packaging, Handling, Storage, and Transportation
- Design Interface

Collectively, these elements illustrate that ILS includes properties of both engineering-oriented and resource-oriented support activities. Hence, supportability is not only a logistical concern but an integrated engineering approach.

Within the scope of this thesis, a relevant connection to the ILS framework lies in Maintenance Planning, where RAM data directly inform long-term maintenance support decisions. The elements Technical Data and Computer Resources Support are also relevant due to the structured and digital nature of the proposed solution, which relies on systematic collection and processing of operational and maintenance data. Finally, Supply Support is indirectly

affected, as reliability performance influences spare part demand, replenishment requirements, and overall support resource consumption.

The Evolution to Integrated Product Support

As mentioned, the traditional ILS framework structures life-cycle support around ten elements. This has evolved to the concept of IPS [31], [32]. While maintaining the same fundamental objectives, IPS expands the ILS element structure to twelve elements by formally introducing Product Support Management (PSM) and Sustaining Engineering (SE) [33]. PSM emphasises governance and coordination of support activities across the life cycle, while SE addresses the continuous assessment and improvement of system performance during the operational phase [33].

The formal inclusion of SE reflects an increased emphasis on in-service performance monitoring, reliability improvement, and continuous support optimisation [33]. This development corresponds with broader life-cycle governance approaches in which operational performance data and structured feedback mechanisms are considered essential for maintaining readiness and controlling LCC [31].

Within the IPS framework, SE therefore formalises the role of life-cycle feedback and operational data analysis. This perspective directly aligns with the objectives of the digital RAM tool developed in this thesis, which focuses on the structured collection and analysis of RAM data during the operational phase.

For the remainder of this report, the terms ILS and IPS are used interchangeably to denote the integrated logistic life-cycle support framework.

IPS in the Context of RAM and In-Service Data

Within IPS, supportability must be evaluated and controlled using measurable performance indicators. RAM attributes constitute central performance measures within the IPS framework, as they influence system readiness and the scope of required support resources [26]. These parameters provide a way to

make informed assessments of whether support strategies meet operational and economic requirements.

During operation, collected in-service data can be analysed to observe reliability performance and identify recurring issues [9]. The S-Series IPS specifications provide standardised structures for managing and exchanging support-related information across the system life cycle [32].

3.1.3 Reliability, Availability, and Maintainability

RAM is commonly treated as a combined set of system attributes as they together describe both the likelihood of failure-free operation and the ability to restore functionality when failures occur [2]. Reliability describes the probability that a system performs its intended function without failure for a specified time under stated conditions [2]. Maintainability captures how quickly and effectively a system can be returned to service after a failure or required maintenance, given defined resources and procedures [2]. Availability expresses the degree to which the system is operational when needed, reflecting the combined effects of reliability and maintainability as well as the operational context in which the system is used [2]. Several variants of the RAM concept are used in the literature and in industry, e.g. R&M, RAMS, RAM-C, and RAMCT. However, this thesis adopts the RAM perspective and focuses on its three core elements, which is explained in more detail below.

Reliability

The reliability of an item (ranging from a single component to an entire system) is commonly defined as the probability that it will perform a required function, under stated conditions, for a stated period of time [34], [35]. In practical terms, this probability is determined by how frequently failures occur during operation [35]. Reliability is therefore strongly dependent on the concept of a failure and, in particular, on how failure is defined and applied in the analysis. This thesis will strictly use the definition “non-conformance to some defined performance criterion” [34]. What constitutes the relevant performance criterion may vary

across applications and must therefore be specified for the situation at hand in order to standardise what is meant by a failure. In practice, reliability work may distinguish between several related terms, such as defects, malfunctions, faults, and rejects, but when assessing an item's reliability it is essential to treat all failures consistently according to the same definition to obtain a robust and comparable reliability analysis [34].

In practice, reliability is often quantified using parameters such as the failure rate $\hat{\lambda}$, Mean Time To Failure (MTTF), and Mean Time Between Failures (MTBF) [34]. From field data, $\hat{\lambda}$ can be estimated as the number of observed failures divided by the total operating time, see Formula 3.1 [34]. However, this estimate represents an average over the observed period and should only be interpreted as a failure rate when the failure rate can reasonably be treated as constant [34]. MTTF is obtained by dividing the total operating time by the number of failures for non-repairable items see Formula 3.2 [34]. For repairable items, MTBF is typically interpreted as the mean up time between successive failures see Formula 3.3 [34].

$$\hat{\lambda} = \frac{k}{T} \quad (3.1)$$

$$\text{MTTF} = \frac{T}{k} \quad (3.2)$$

$$\text{MTBF} = \frac{T}{k} \quad (3.3)$$

These measures can be computed over different time windows, rather than over an item's entire life cycle, which is useful for revealing how failure behaviour changes over time and for distinguishing between different failure types. A common way of illustrating this is the bathtub curve, where early-life failures ("infant mortality"), a useful-life period with approximately constant failure rate ("random failures"), and wearout failures correspond to different underlying failure mechanisms and therefore require different analytical treatment [34].

Figure 3.1 presents a typical example of a bathtub curve for an item.

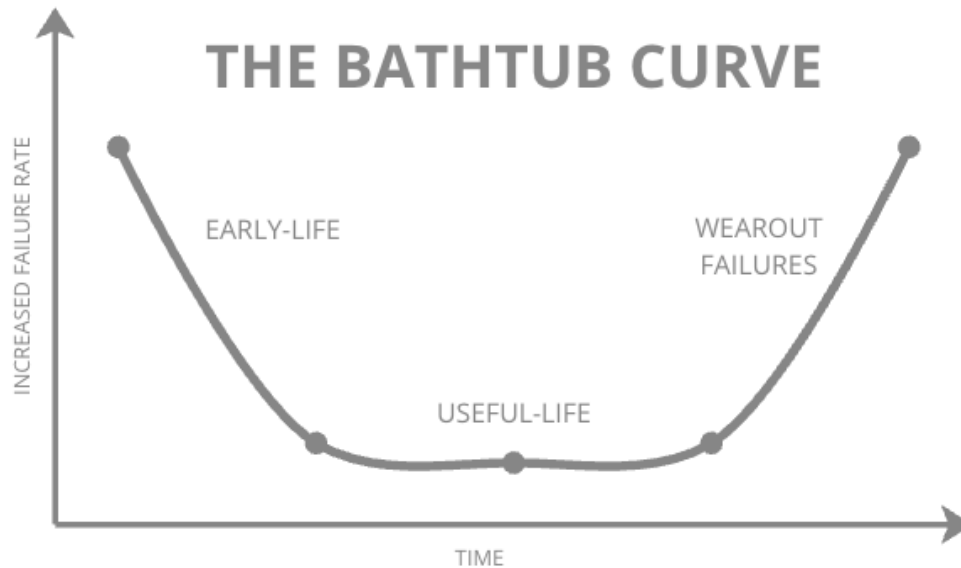


Figure 3.1: Bathtub curve illustrating a typical early-life, useful-life, and wear-out failure behaviour.

Maintainability

Maintainability can be defined as the probability that a failed item can be restored to an operationally effective (i.e. repaired) condition within a specified period of time, typically by performing corrective maintenance in accordance with prescribed standard procedures [35]. This definition highlights that maintainability is not only about the ability to restore function through corrective action, but also about doing so within an acceptable time frame and with suitable procedures.

As for reliability, there are different time-based metrics to quantify maintainability of an item (component or system). The key measures used, which are often confused with each other, are Mean Time To Repair (MTTR) and Mean Down Time (MDT) [34]. Down time or outage can be described as the period during which equipment is in a failed state [34]. Providing a more formal definition is avoided due to the difficulty of generalising a parameter whose elements

may vary depending on the system and its operating conditions [34]. MTTR is the mean active corrective maintenance time, including diagnosis, repair, checkout, and alignment where applicable, estimated as total active repair time divided by the number of corrective actions, see Equation 3.4 [34], [36]. MDT represents the mean total time the equipment is unavailable, estimated as total accumulated downtime divided by the number of outages, see Equation 3.5 [34], [36]. These measures overlap but are not always identical, since down time may begin before repair starts and repair activities (e.g. checkout or alignment) may extend beyond the outage [34].

$$\text{MTTR} = \frac{\sum_{i=1}^{N_r} t_{\text{repair},i}}{N_r} \quad (3.4)$$

$$\text{MDT} = \frac{\sum_{i=1}^{N_o} t_{\text{down},i}}{N_o} \quad (3.5)$$

Availability

Availability describes how often a system is ready and able to perform its intended function when required, and it can be expressed as the fraction of time the item (again, component or system) is in an operational state [35]. In practice, availability is assessed through measures that connect how frequently failures occur with how long recovery takes [35], that is the time-based metrics for reliability and maintainability described above (MTBF together with MDT). In steady state, availability can be expressed as the ratio of up time to total time, which then can be expressed in terms of the mentioned metrics, see Equation 3.6 [34].

$$A = \frac{\text{Up time}}{\text{Up time} + \text{Down time}} = \frac{\text{MTBF}}{\text{MTBF} + \text{MDT}} \quad (3.6)$$

Another way of analysing the availability, which can be more convenient in some applications, is to instead look at the unavailability, which is also known as the probability of failure on demand (PFD), see Equation 3.7 [34].

$$\bar{A} = 1 - A = \frac{\lambda \text{MDT}}{1 + \lambda \text{MDT}} \approx \lambda \text{MDT} \quad (3.7)$$

3.1.4 RAM Prediction and Statistical Estimation

RAM prediction (or RAM parameter estimation) is largely concerned with estimating RAM-related parameters from test or field data, where the available observations may be complete, incomplete, or censored [35]. These estimation activities are typically formulated in probability theory and rely on choosing an appropriate parametric distribution model for the observed data [35]. Because estimated parameters will vary across datasets even when the item is unchanged, the estimate should be interpreted together with a measure of significance, commonly expressed as a confidence level [35]. In this context, parameter estimation and uncertainty quantification are usually supported by methods such as maximum likelihood estimation and Bayesian estimation [35].

A particularly important baseline model for reliability prediction in the useful-life region is the exponential (Poisson) failure distribution, which assumes a constant failure rate and therefore yields a simple analytical reliability function [35]. Equation 3.8 shows the expression of the probability density under the exponential model [35].

$$f(t) = \lambda e^{-\lambda t} = \frac{1}{\text{MTBF}} e^{-t/\text{MTBF}} \quad (3.8)$$

Further, λ can be estimated either graphically via probability plotting or analytically using least squares or maximum likelihood [35]. This makes the exponential model attractive when data are limited and interpretability is prioritised.

When failure behaviour is not well represented by a constant hazard rate, more flexible distribution models may be used to capture time-varying failure behaviour. One of these is the Weibull distribution, which provides a more flexible alternative because its shape parameter changes the implied failure-rate behaviour over time [35]. In the two-parameter Weibull model, the probability

density function is given by Equation 3.9 [35]. Equation 3.10 provides the corresponding reliability (survival) function and Equation 3.11 shows the associated failure rate (hazard) function [35]. The reliability function gives the probability that an item survives without failure up to a certain time, while the failure rate function describes the instantaneous propensity to fail at a certain time given that the item has survived up to that time, indicating whether failure risk decreases, remains approximately constant, or increases over time [35].

$$f(t) = \frac{\beta}{\eta} \left(\frac{t}{\eta} \right)^{\beta-1} \exp \left[- \left(\frac{t}{\eta} \right)^{\beta} \right] \quad (3.9)$$

$$R(t) = \exp \left[- \left(\frac{t}{\eta} \right)^{\beta} \right] \quad (3.10)$$

$$h(t) = \frac{f(t)}{R(t)} = \frac{\beta}{\eta} \left(\frac{t}{\eta} \right)^{\beta-1} \quad (3.11)$$

The shape parameter β governs whether the hazard is decreasing, constant, or increasing, while the scale parameter η sets the characteristic life level of the distribution [35]. Weibull probability plotting (often referred to as a Weibull plot) is widely used as a graphical method for assessing whether time-to-failure data can reasonably be modelled by a Weibull distribution [35]. By sorting the observed failure times and assigning cumulative failure probabilities using median ranks, commonly approximated by Benard's approximation [37], the data can be plotted on Weibull-linearised axes. If the data are reasonably represented by a Weibull distribution, the plotted points will approximate a straight line, providing a visual check of model adequacy [35]. Figure 3.2 illustrates this principle. The slope of this line corresponds to the Weibull shape parameter β , while the position of the line is used to determine the scale parameter η [35]. Because β indicates whether the failure rate is decreasing, approximately constant, or increasing, the Weibull plot provides an interpretable link between observed failure patterns and the underlying failure behaviour across different life-cycle phases [35].

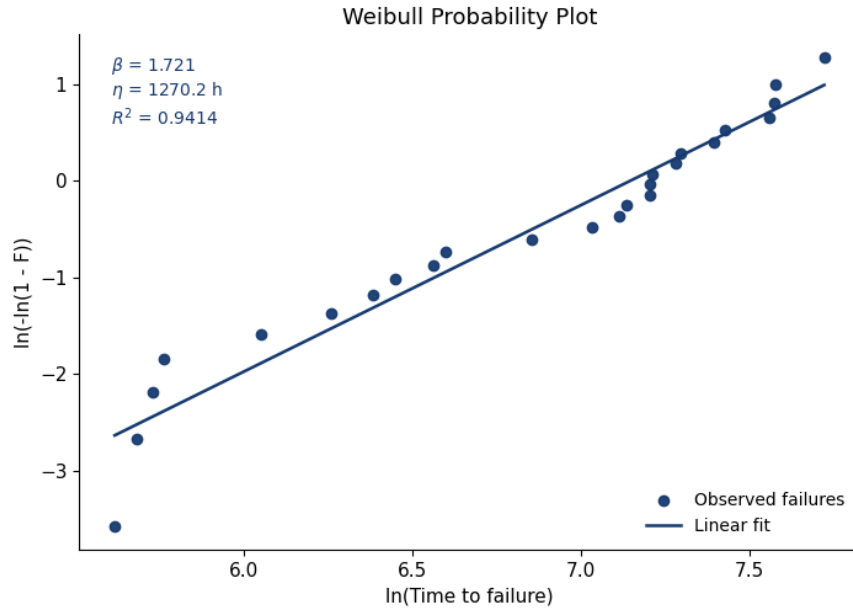


Figure 3.2: Weibull probability plot based on mock time-to-failure data. The plotted points represent observed failures, while the fitted line shows the linearised Weibull relationship used to estimate the shape parameter β and the characteristic life η . The coefficient of determination, R^2 , indicates the goodness of fit.

In RAM analysis, this is relevant because the chosen distribution and parameter values directly influence predicted reliability measures and associated confidence bounds [35]. The Weibull plot is also practically useful during early assessments because it supports graphical parameter estimation and can provide a broad indication of the type of distribution represented by the failure data, even when relatively few failures are available [35]. Deviations from a linear pattern may also prompt further investigation of the data, for example whether different failure mechanisms or inconsistent failure definitions are present.

For repairable systems, prediction typically needs to combine both time-to-failure and time-to-repair behaviour, since availability depends on failure and repair processes rather than a single event process [35]. Therefore, assigning confidence levels to availability is not straightforward to do parametrically,

hence simulation approaches such as Monte Carlo simulations are commonly employed based on estimated time-to-failure and time-to-repair distributions [35]. At the system level, availability can be modelled by defining operational and failed/repair states and analysing how the system moves between them over time, enabling derivation of measures such as steady-state availability and mean time to failure under stated failure and repair assumptions. Overall, the choice of statistical model and estimation approach (e.g. exponential or Weibull) directly affects the resulting predictions and their uncertainty, and for repairable systems the same principles extend to availability by combining failure- and repair-time behaviour at the system level.

3.1.5 Reliability Growth and Trend Analysis

Reliability growth describes how the reliability of a system improves over time as failures are discovered and addressed through corrective actions during development and test [7]. A central idea in reliability growth management is that observed failure-event data from a test phase can be used both to track whether growth is occurring, and to project reliability at later points in the same phase, provided that the assumptions about corrective actions are stated [7]. In practice, reliability growth analysis is therefore often used as a quantitative complement to FRACAS by turning reported failures and implemented fixes into trend measures that indicate whether the failure occurrence rate is decreasing over time [7].

A common empirical approach is the Duane model, where the cumulative mean time between failures follows a learning-curve relationship with cumulative test time [7]. Equation 3.12, where t is cumulative test time, C is a constant, and α is the growth rate, shows how this relationship commonly is expressed [7].

$$\text{MTBF}_c(t) = Ct^\alpha \quad (3.12)$$

A Duane plot is obtained by plotting $\log(\text{MTBF}_c)$ against $\log(t)$, where

approximately linear behaviour supports the model assumption and the slope provides an estimate of α [38]. Interpreting the slope is straightforward: $\alpha > 0$ indicates improving reliability (increasing cumulative MTBF), while $\alpha \approx 0$ indicates little or no growth over the analysed interval [38]. Figure 3.3 shows an example of a Duane plot.

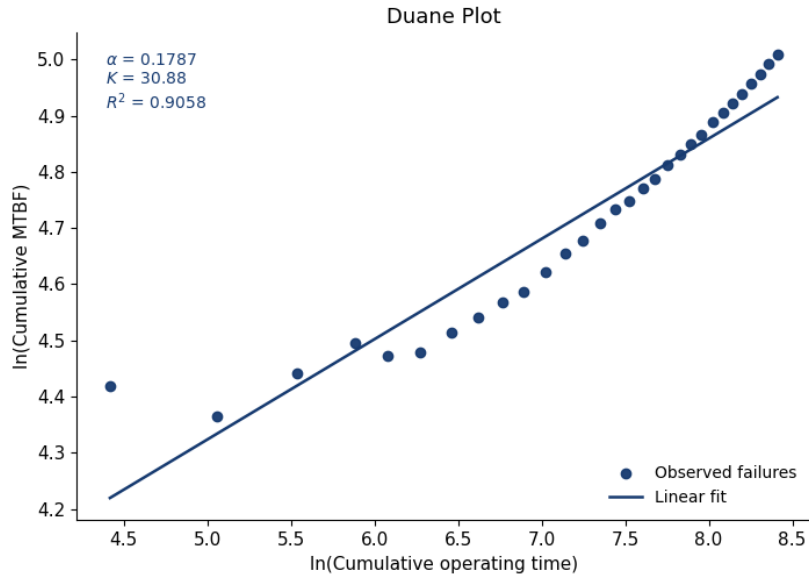


Figure 3.3: Duane plot based on mock failure data for a repairable system. The plotted points represent the relationship between cumulative operating time and cumulative MTBF in logarithmic form, while the fitted line shows the linearised Duane model used to estimate the growth parameter α . R^2 indicates the goodness of fit.

A widely used statistical framework for reliability growth is the Crow-AMSAA model, which represents failure occurrences during a test phase as a power-law non-homogeneous Poisson process (NHPP) [7]. Equation 3.13, where λ and β are parameters estimated from the observed failure-event times, shows the expected cumulative number of failures by time t in this model [7].

$$M(t) = \lambda t^\beta \quad (3.13)$$

The parameter β supports interpretation of the trend in failure intensity over time, where values $\beta < 1$ indicate a decreasing failure intensity (growth), $\beta = 1$ corresponds to an approximately constant failure intensity, and $\beta > 1$ indicates an increasing failure intensity over the analysed period [7]. Because the model is based on failure-time data within a defined test phase, it is well suited when individual failure times, or grouped failure times, are available and the objective is to quantify reliability trends during development testing [7]. This is particularly relevant in test-fix-test settings, where corrective actions are implemented during the test phase [7].

Both Duane and Crow-AMSAA are commonly used to support engineering decision-making during test programmes, for example by indicating whether corrective actions are producing the desired trend and by providing a basis for reliability projections under stated assumptions [7]. In a FRACAS-supported workflow, the practical implication is that the failure-event data must be traceable to time/exposure and configuration, and corrective actions should be traceable in time so that trend changes can be interpreted in relation to what was actually implemented [7].

3.1.6 Non-Parametric Approaches for RAM

In addition to parametric lifetime and reliability-growth models (e.g. Weibull and Duane/Crow-AMSAA), RAM analysis can be complemented by data-driven time-to-failure modelling where the survival function (and related reliability measures) is estimated directly from operational data without assuming a specific lifetime distribution [39]. In predictive maintenance settings, such approaches are attractive because they naturally support censored observations, including cases where the exact failure time is unknown because the observation period ends before failure occurs or because the unit remains in operation without an observed failure [39], [40]. From a RAM perspective, the resulting survival or reliability estimates can be used to inform maintenance timing and prioritisation, which in turn affects downtime and availability [41]. Recent work in the prognostics and health management (PHM) literature also

highlights that “survival-analysis style” models provide an alternative to pure regression-based remaining useful life (RUL) prediction, since both approaches address time-to-event prediction but originate from different research traditions [42]. Survival methods are particularly relevant in this context because they explicitly model time-to-event behaviour and can incorporate censored observations, which are common in maintenance datasets [39], [41]. Overall, these non-parametric and machine-learning-based time-to-event approaches can therefore serve as a practical complement to Weibull and reliability-growth models when field data are heterogeneous and when the primary goal is to support RAM-relevant decisions in operation and sustainment [39], [41]. While these data-driven and non-parametric approaches are promising, they fall outside the scope of this thesis, as a meaningful application would require substantially more work to collect, clean, and validate sufficiently rich operational datasets, as well as to develop and evaluate suitable models under the constraints of the target environment.

3.1.7 Reliability and Safety Analysis Methods

Reliability and safety engineering commonly relies on a set of complementary methods that support both systematic identification of failure mechanisms and structured follow-up through corrective actions [43]. In this context, Functional Hazard Analysis (FHA), Failure Mode, Effects and Criticality Analysis (FMECA), and Fault Tree Analysis (FTA) are primarily used to identify hazards, failure modes, and causal relationships, while the Failure Reporting, Analysis and Corrective Action System (FRACAS) provides a closed-loop process for reporting failures, analysing their causes, and performing corrective actions [43]. Because these methods address different questions and system levels, they are often applied together to improve analytical coverage and maintain traceability from hazards and failure causes to implemented corrective actions [43]. This subsection introduces these methods and outlines how they can be integrated into a coherent reliability and safety workflow rather than treated as isolated analyses. It also highlights reliability analysis as a quantitative way to monitor whether reliability improves over time as

corrective actions are introduced.

Failure Reporting, Analysis and Corrective Action System

FRACAS is commonly described as a closed-loop process that links failure reporting, root-cause analysis, and corrective actions in order to prevent recurrence and to support reliability and maintainability improvement over time [43]. Figure 3.4 illustrates the closed-loop nature of FRACAS. Within defence acquisition guidance, FRACAS is framed as a disciplined mechanism intended to provide management visibility and control by using failure and maintenance data to generate and implement effective corrective actions, both to prevent recurrence and to simplify or reduce maintenance tasks [6]. A key implication is that FRACAS is not limited to “logging failures”, but aims to ensure that each reported failure is analysed, assigned an attributable cause, and followed through to an implemented resolution [6].



Figure 3.4: Illustration of the FRACAS closed-loop process.

From a process perspective, FRACAS typically includes the following elements [6]:

- i) failure reporting,
- ii) failure analysis,
- iii) failure verification,
- iv) corrective action,
- v) failure report close out,
- vi) identification and control of failed items.

To ensure oversight and prioritisation, an organisational review function is often established (e.g. a Failure Review Board) to monitor failure trends and corrective-action status and to drive timely decisions [6]. Effective FRACAS execution depends on maintaining records that connect reported failures to

analyses, assignable causes, corrective actions, and verification evidence, using identifiers that enable efficient retrieval and traceability across the workflow [6]. Timely close-out is emphasised to avoid an unmanageable backlog of open reports, where close-out implies that corrective actions have been implemented and verified, or that a documented and approved rationale exists when corrective action is not implemented [6].

In practice, FRACAS can be viewed as the operational “feedback loop” that complements up-front analytical methods such as FTA and FMECA, since test and field failures provide evidence that can refine assumptions, update identified failure modes, and re-prioritise risk drivers [43]. To avoid duplication and achieve consistent outcomes, FRACAS is typically coordinated with related activities such as reliability, quality assurance, maintainability, system safety, configuration management, and ILS [6]. These characteristics are particularly relevant for this thesis because they translate directly into information and workflow requirements, including structured failure reporting, status tracking for corrective actions, and traceable verification of effectiveness.

Other Reliability and Safety Analysis Methods

In addition to FRACAS, reliability work commonly relies on a set of structured methods of analysis that support identification of failure causes, prioritisation of risk, and modelling of system-level behaviour [43]. In the preliminary design phase, these methods can include mathematical modelling approaches such as Markov modelling (to estimate reliability for multi-state systems under constant failure and repair rates of independent assemblies), the binomial method (to assess reliability for simple series and parallel configurations), and equipment ageing models (to represent degradation processes at varying rates in engineered installations) [35]. As earlier mentioned, complementary cause–effect methods are also widely used, for example FMEA/FMECA, which assess failure effects and criticality in a step-by-step manner, and FTA which uses a deductive top-event logic to identify the causal relationships and event combinations that can lead to a specific system failure mode [35]. FHA is commonly used to evaluate hazards at the functional level and to link loss

or degradation of functions to operational consequences, which makes it particularly relevant when RAM work is tied to safety- or mission-critical performance [43].

A practical distinction between these methods is the perspective they impose on the problem formulation. FMEA/FMECA is typically applied bottom-up by examining each item or function, enumerating credible failure modes, and tracing their local and system effects, which supports early identification of critical weak points and documentation of design assumptions [35]. FTA, by contrast, is applied top-down by defining an undesired system event and structuring the logical paths that can cause it, which is useful when the analysis needs to demonstrate how particular failure modes and dependencies can lead to a system-level loss [35]. Markov modelling and related state-based approaches emphasise time dependency by modelling transitions between operational, degraded, and failed states, which make them suitable when redundancy, repair, and restoration behaviour are central to the predicted RAM performance [35]. From a data perspective, these approaches also imply different information needs [35]. Cause-effect methods require consistent failure mode and consequence descriptions, whereas state-based models additionally require assumptions (or data) for transition rates such as failure and repair rates [35]. This makes the methods not only analytical techniques but also drivers of how reliability and maintenance information should be structured in order to support credible assessments across the system life cycle [43].

Combinations of Multiple Reliability and Safety Analysis Methods

In practice, these methods are often applied in coordinated “method sets” rather than as stand-alone activities, where the combination is selected based on the analysis purpose and the system life-cycle phase [43]. Zhang et al. describe common integration patterns in terms of increasing levels of coverage, ranging from two-framework (2F) to three-framework (3F) and four-framework (4F) combinations [43]. A typical 2F approach pairs methods that complement each other’s perspective, such as combining a bottom-up technique (e.g. FMECA) with a top-down technique (e.g. FTA) to connect component-level failure modes

to system-level undesired events [43]. Extending to a 3F approach commonly introduces FRACAS to provide a closed-loop mechanism for collecting failure evidence and tracking corrective actions, thereby enabling analytical models to be updated and validated against operational experience [43]. In a 4F approach, FHA is often added to strengthen the functional and hazard-oriented view, improving traceability from functional hazards to system failure logic and to corrective-action follow-up [43]. Overall, combining these methods helps to maintain consistency and traceability across functions, failure modes, system events, and corrective actions, while reducing the risk of blind spots that can occur when relying on a single method alone.

3.1.8 S5000F - International Specification for In-Service Data Feedback

This subsection presents the findings of the literature review of the S5000F specification. Firstly, it introduces the S-Series suite as a whole, to provide conceptual context for S5000F. Secondly, it explains the in-service data feedback process and how the specification relates to it. Thirdly, an overview of the specification and its intended use is presented, together with its recommended tailoring process.

The S-Series Suite of IPS Specifications

The S-Series suite of IPS specifications is a set of standards, constituting a framework for managing the support and maintenance of systems over their entire life cycle [44]. Its goal is aligned with that described for ILS, to ensure that complex systems remain supportable, effective and reliable [44]. The suite is governed by the Aerospace, Security and Defence Industries Association of Europe (ASD) and the specifications have been developed in participation with an international team of experts from other organisations, where Saab is included [9], [45].

The suite defines a structured Common Data Model (CDM) and business rules intended to enable interoperability between organisations and information systems involved in IPS [9]. An overview of the international specifications

included in the suite can be viewed in Table 3.1.

Table 3.1: All S-Series suite specifications

Title	International Specification for:
S1000D	technical publications development
S2000M	material management
S3000L	logistic support analysis process
S4000P	develop. and continuously improving preventive maintenance
S5000F	in-service data feedback
S6000T	training analysis and design
SX000i	integrated product support

Although each specification has its own scope and application area, they are designed to be complementary and interoperable [32]. Together, they aim to enable consistent data structures, traceability, and controlled information exchange across stakeholders throughout a system’s life cycle [9].

Rather than determining how organisations develop their internal software systems, the S-Series is intended to ensure consistent interpretation of support-related data across actors, such as operators, maintainers, and original equipment manufacturers [9]. This is typically achieved through structured data exchange messages in eXtensible Markup Language (XML) format [9], [45]. During data exchange between two actors, these messages are validated against the specifications’ XML Schema Definitions (XSD) [9], [45]. In this context, a schema defines the required XML structure, including allowed elements, attributes, and data types, enabling automated validation and consistent data exchange.

To ensure the needed consistency in interpretation of exchanged data, the suite provides support specifications depicted in Table 3.2 [46]. SX001G provides a standardised glossary of terms and definitions applicable across the specifications, supporting consistent communication throughout the suite [46]. SX002D defines the CDM forming the foundation of the specifications, providing data structures and relationships to ensure consistency and interoperability [46].

SX004G serves as a reader’s guide to the Unified Modeling Language (UML)-based data models used, facilitating correct understanding and application of the CDM [46]. SX005G provides guidance for the implementation and application of XSDs within the S-Series framework, supporting consistent validation [46].

Table 3.2: All S-Series suite support specifications

Title	Specification Purpose
SX001G	Glossary
SX002D	Common Data Model
SX004G	UML Model Reader’s Guide
SX005G	XML Schema Implementation Guidance

Position of S5000F Within the IPS Information Flow

S5000F is the S-Series IPS specification governing the structured collection and exchange of in-service data feedback [9]. In this context, in-service data feedback refers to operational and maintenance-related information generated during system use. These data are transferred back to systems engineering, logistics, and support functions to enable data-driven decision-making. Its primary objective is to define a framework for capturing, structuring, and transferring data generated during the operational phase of a product’s life cycle [9]. The aim of using in-service data feedback is to improve product availability and improve product LCC by enabling informed decision-making based on operational performance data [9].

S5000F may be used alone or together with other S-Series IPS specifications [9]. Either way, it provides a framework for handling data feedback from actors, such as operators, maintainers, and contractors [9]. Using S5000F together with other specifications, in-service data may contribute to updates of logistic support analyses under S3000L, refinement of preventive maintenance tasks under S4000P, or improvements of material management under S2000M [9]. In this sense, S5000F functions as a feedback interface within the IPS information

architecture, linking operational and maintenance data to analytical, planning, and documentation processes.

A schematic overview of the S-Series specifications' positions in the information flow and their respective areas can be viewed in Figure 3.5. From the figure, one can see that S5000F provides data to all other specifications, enabling informed decision-making.

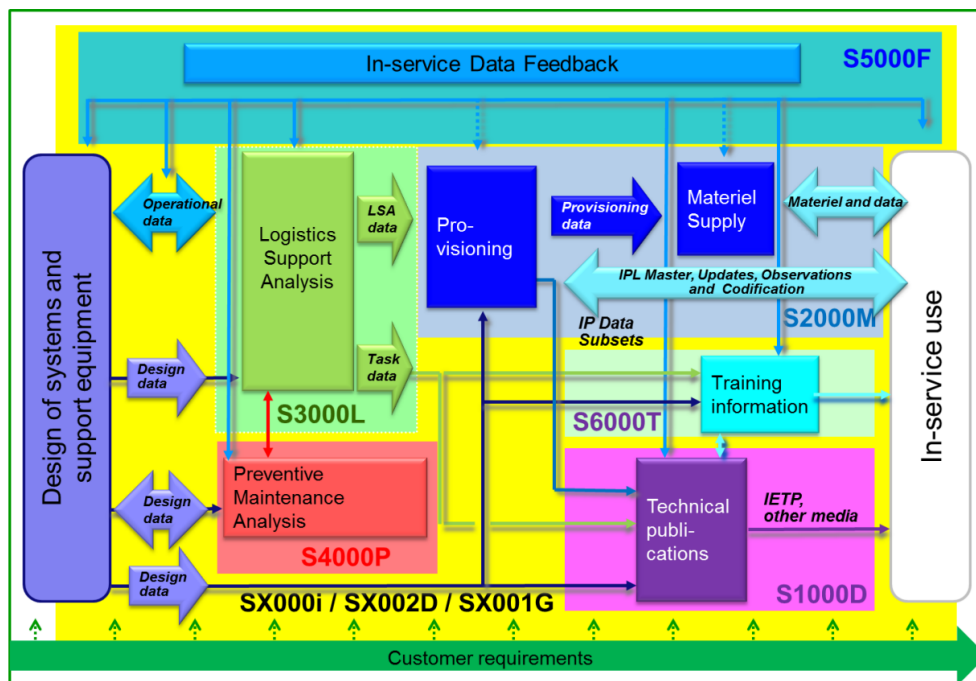


Figure 3.5: Schematic overview of the S-Series specifications. Adapted from [9].

As mentioned, S5000F can be used on its own, without implementing the entire S-Series suite [9]. Figure 3.6 shows how S5000F is positioned right between two actors, governing in-service feedback data exchange.

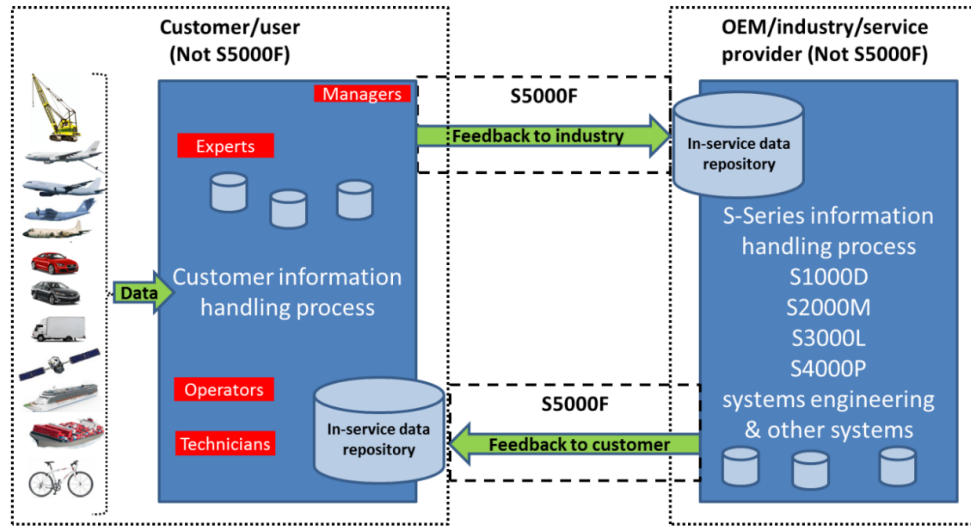


Figure 3.6: Overview of the role of S5000F between actors. Adapted from [9].

In the context of this thesis, data exchange between the digital tool and other internal systems might be of interest. However, data exchange with external stakeholders, such as customers, remains outside the scope of this study. Referencing Figure 3.6, the developed tool could be positioned as the left-hand entity (Customer/User), where data collection occurs independently of S5000F. On the other hand, the data exchange to other internal tools, represented by the right-hand entity (OEM/Industry/Service Provider), is governed by the S5000F standard.

Scope and Limitations of S5000F

S5000F does not mandate the design of internal IT systems, as its scope is limited to defining standardised information exchange messages between actors [9]. Consequently, the specification neither dictates the use of a single or multiple in-service databases nor defines their internal data structures [9]. Since the data schemas are highly dependent on system-specific characteristics and the operational information required, such implementation details fall outside the scope of the specification [9]. It is not possible for the specification to describe all variations of the processes related to a product, since they differ

from product to product [9]. For example, some data classes have strictly controlled valid values. If deemed necessary for a project, these values can be reduced or expanded without compromising specification compliance [9]. However, the changes to the XSD must be done by both actors validating the messages utilising project-specific values [9].

Furthermore, S5000F does not define how in-service data are generated [9]. While the specification structures the exchange of operational and maintenance data, it does not regulate how events are detected, recorded, or validated within internal systems [9].

Similarly, how calculations are derived from in-service data, e.g. MTBF, falls outside the scope of S5000F [9]. Although such metrics may rely on data structured according to the specification, the corresponding algorithms and calculations are defined within other specifications, e.g. S3000L [9]. S5000F governs the structured feedback of data, but not the analytical interpretation of it.

Tailoring and Implementation Considerations

S5000F is intentionally designed to be tailored to the needs of a specific project or feedback objective [9]. Rather than requiring implementation of the entire data model, the specification supports selective adoption of functionality so that only the feedback content relevant to a given operational context is exchanged [9]. The required depth of implementation is therefore determined by the nature of the intended feedback use [9].

This modularity makes it possible for S5000F to be applied contractually between actors [9]. In such a contractual agreement, stakeholders agree on a set of feedback activities (use cases) to be supported, and the corresponding information content is defined by referencing the specification's mappings between use cases and data requirements [9]. While contractual exchange between organisations is a primary driver for this modularity, the same approach can also be used within an organisation to ensure that data capture and export are aligned with a defined and traceable information need [9].

Tailoring in S5000F can be described as a selection process in which the intended feedback purpose determines the exchanged information content. At a conceptual level, this process consists of the following steps:

1. Define the feedback need and select relevant use cases.

S5000F relates feedback data to typical use cases, enabling a project to start from the intended application of the data (e.g. reliability monitoring, maintenance analysis, or supply support for spares management) [9]. This selection establishes the reason why information should be exchanged.

2. Identify the corresponding Units of Functionality.

Use cases are mapped to Units of Functionality (UoF), which split the overall data model into subsets of data models needed to support specific exchange scenarios [9]. UoFs act as intermediate building blocks between high-level feedback objectives and the detailed information entities required for exchange, enabling a user of the specification to more easily understand the complete data model [9]. There are both mandatory and optional UoFs for each use case, but even the mandatory ones are recommendations by the specification [9]. Hence, the user must determine whether all UoFs are applicable for a specific information exchange.

3. Derive the relevant information entities from the selected UoFs.

Each UoF references classes within the underlying data model [9]. These classes define the entities, attributes, and relationships that are the basis of the information exchange [9]. During tailoring, a project determines which classes provide necessary information for the selected use cases and which can be excluded without compromising the intended feedback purpose [9].

4. Formalise the exchange message definition.

S5000F provides XSDs to support validated compliance [9]. The baseline schema is intentionally permissive to accommodate variations of exchange agreements, while project-specific definitions determine the agreed information to be exchanged without modifying the CDM [9]. This separation supports S-Series interoperability by preserving the baseline specification while allowing

stricter constraints to be applied at a project-specific level [9].

S5000F notes that feedback datasets linked to different use cases may overlap, meaning that the same data elements can be relevant in multiple contexts [9]. This overlap increases the complexity of data modelling, as information must be structured in a way that supports reuse while maintaining clear traceability between feedback purpose and data content [9]. The stated tailoring approach described above therefore helps ensure that selected data elements are explicitly linked to defined use cases, providing both scope control and traceability [9].

3.2 Stakeholder Interviews

In addition to the literature study, stakeholder interviews were used as an empirical input to the requirement elicitation process. The interviews aimed to capture practical insights into how RAM data are currently handled within the organisation, as well as the requirements and expectations of the digital RAM tool. Particular attention was given to understanding both the analytical use of RAM data and the practical conditions under which operational data could be reported and documented in everyday work.

3.2.1 Stakeholder Roles and Interview Participants

According to the methodological approach described earlier, the intention was to interview two main categories of stakeholders: RAM/ILS experts representing the analytical or output-oriented use of the tool, and operational personnel representing the input side, i.e. users responsible for reporting failures, maintenance actions, and other operational events into the system.

In practice, however, these roles were found to overlap significantly within the organisation. Many stakeholders involved in RAM and ILS activities also possess operational insight and may both contribute data and use the resulting analyses. Consequently, the distinction between input and output users was not always clear-cut. For example, RAM/ILS personnel may both analyse reliability data and participate in the reporting or validation of operational

events.

In addition to the previously described interviewees, relevant IT personnel were interviewed to ensure compliance with internal best practices, as well as to understand the IT environment and its underlying constraints. An overview of the conducted interviews, including the interview format and the stakeholder roles represented in each interview, is presented in Appendix C.

3.2.2 Use Context and Applications

The interviews suggest that the proposed tool may support a broad range of use contexts within the organisation. Different stakeholders are expected to engage with the tool in different ways, reflecting varying objectives, responsibilities, and information needs. One identified application is in development projects, where access to historical data from similar systems could support comparisons and more informed engineering and design decisions [I1]. Stakeholders have also highlighted the value of having structured access to data that enables RAM analyses across different contexts and for different purposes, such as comparing current outcomes with previous analyses and earlier product generations to assess whether a product performs in line with targets and predictions [I1, I3].

Another anticipated use concerns the comparison of individual units within a product, which could help identify patterns and generate insights that may, for example, motivate proposals for product or process improvements to management [I2]. In addition, results produced through the tool may provide supporting input for updates to manuals and other technical documentation [I4]. Taken together, these findings indicate that the tool will not be associated with one single use case, but rather with a set of analytical and decision-support functions. This makes it important to account for multiple user perspectives and requirements of the tool design.

3.2.3 Analytical Needs and Outputs

As stakeholders are expected to use the tool for different purposes, they also expressed interest in different types of outputs and analytical support. Different

analyses require different measures, calculations, and forms of presentation. One important expectation was that the tool should handle the full process through which a failure report is received, processed, analysed, and eventually addressed, thereby creating a more complete picture of how issues develop over time [I1]. In this sense, stakeholders highlighted the value of being able to follow a product throughout its operational life and to connect incidents or failure reports to subsequent analyses, decisions, and corrective actions [I1, I4].

To support such use, both relatively simple and more advanced analytical measures were considered relevant. Common reliability-related indicators such as MTBF and the equivalent for distance, Mean Distance Between Failures (MDBF), were mentioned as important baseline measures, while more advanced models and visualisations, such as reliability analysis, including shape and scale parameters, through Weibull plots and reliability growth analysis through Duane plots, were seen as useful for deeper analytical purposes [I1]. One particularly clear expectation was the ability to assess whether products perform in accordance with what was stated or assumed at the time of sale, for example whether predicted reliability levels have been achieved in practice, and to identify when corrective actions may be needed if this is not the case [I3].

In addition, stakeholders emphasised the importance of enabling comparisons, for example, with regard to the number and types of failures, both between individual units and across products [I2]. This implies that reported events need to be classified at a sufficiently detailed level to allow meaningful grouping, filtering, and comparison. It was also considered valuable to present failures according to factors such as frequency of occurrence, in order to support the identification of recurring issues and relevant patterns [I2]. This could include functionality for flagging an individual unit or product if a certain number of failure reports are recorded within a given period of time, as well as the ability to display the proportion of no fault found (NFF) cases within different problem categories [I1]. Here, NFF refers to cases in which a reported fault cannot be confirmed during inspection, testing, or troubleshooting, despite the initial indication that a failure has occurred.

The reasons for requesting these outputs vary between stakeholders, but in several cases the purpose is to generate information that can serve as a basis for proposed actions, either internally towards management or externally towards customers [I2, I3]. For this reason, it may be beneficial if the tool supports convenient production and extraction of presentation material. At the same time, it was noted that while formal export of values or graphs could be a useful feature, it was not seen as a requirement, since stakeholders do not emphasise the need for formal referencing in that context [I1, I2].

Some stakeholders also expressed a need for export functionality, as in the ability to export selected data from the tool either for import into another internal system or for further use in other contexts. The specific data of interest appears to depend on the stakeholder in question, and some stakeholders expressed no need for export functionality at all. However, the export of action reports was discussed as a relevant use case, and in some cases also the possibility of exporting analysis metrics such as MTBF, MTTR, and related measures [I1]. In this case, XML was mentioned as a relevant format to ensure compatibility with other systems and compliance to S5000F, which is something that Saab as an organisation has decided to work towards [I1, I2]. The primary interest was not to export raw collected data directly, but rather selected analytical outputs, such as calculated reliability measures, for example MTBF and availability values [I1].

3.2.4 Input Requirements and Data Structure

The interviews indicate that defining data requirements is not entirely straightforward, since both the availability and feasibility of collecting specific data may vary significantly across customers and operational contexts [I1]. One implication of this is that the design should not be based solely on the data that is currently easiest to access, but rather on the information considered necessary for the intended analyses, within reasonable practical limits [I1].

Several types of information were nevertheless identified as particularly important. Data describing where, when, and how a failure occurred was consistently

regarded as relevant, both for the classification and analysis of the reported events [I4]. To enable meaningful comparison between failures, the input data must therefore support structured categorisation at multiple levels, both by product and by failure characteristics. This includes classification of an incident by discipline (such as electrical/electronic, mechanical or software/firmware-related faults), which customer who owns the unit, and who reported the issue [I1, I2]. Relevant dates are also necessary data to collect, both when the error occurred and when it arrived at and left the workshop [I1]. This is relevant because both the time spent resolving the issue and the total period during which the unit was unavailable are important for the analysis. The input should also be capable of being complemented by estimated cycle counts or travelled distance for components where such measures provide a more meaningful basis for analysis [I1]. In this context, cycle counts refer to the number of operating cycles a component has undergone before failure. This measure is relevant for products or components for which cycles provide a more meaningful basis for analysis than elapsed time, such as a button, where each press constitutes one cycle.

In addition, stakeholders identified several further data elements as relevant for describing the nature of a reported failure. These include the outcome of a root cause analysis (RCA), whether the case should be classified as NFF, categorisation of the root cause, a description of the observed symptoms, and the troubleshooting actions that were undertaken [I1]. Stakeholders also described a need to connect failures to, for example, components, failure area, individual units, and product [I2]. Such a structure would support both traceability and comparison across different levels of analysis. In practical terms, this requires the inclusion of identifiers for components, incidents, and individual units [I2, I4].

3.2.5 Adoption and Acceptance Criteria

Beyond the question of which specific functions and data the tool should include, the interviews also highlighted several broader conditions that are

important for the tool to be adopted and regarded as useful in practice. One aspect that recurred across more or less all interviews was usability. Some interviewees identified ease of use and an intuitive interface as the single most important characteristic of the tool [I1].

The strong emphasis on usability appears to be linked to the current situation, in which existing systems are not used in a uniform or routine-based manner [I1, I3, I4]. This suggests that it is not sufficient for the tool to have a simple and intuitive graphical user interface alone. Organisational support, clear routines, and an established way of working around the system are also needed if it is to be used consistently in practice [I3]. In addition, stakeholders indicated that visible value in everyday use, such as evidence that the tool functions well and produces relevant analyses, could itself serve as an incentive for more regular use [I3].

Another factor highlighted as important was the extent to which the tool fits into the organisation's broader system environment. Some interest was expressed in enabling the transfer of information from other internal systems into the tool [I3]. At the same time, it was recognised that broader integration with such systems would likely fall outside the scope of the thesis and if some data entry from other systems were to be deemed necessary, it would probably need to be performed manually [I1, I3].

3.2.6 Technical Constraints and Design Considerations

The stakeholder interviews provided important insights regarding technical constraints and considerations for the development of the digital RAM tool. These discussions helped establish guiding principles for system architecture, deployment strategy, and long-term maintainability.

The value of enabling continuous deployment, allowing supervisors to continuously access the system in order to evaluate progress and provide feedback, during application development was emphasised [I1, I2]. To achieve this within the organisational constraints, guidance to host the application in a cloud-based environment was provided [I5]. Furthermore, automated deployment through

modern tooling was presented to facilitate rapid iteration, which would enable the desired continuous supervisor validation of the system [I5].

Stakeholders required that the final system must be deployable within a secure internal IT environment, as RAM data often is sensitive and subject to security restrictions [I1, I2]. Hence, the system cannot rely on cloud-native services and should remain capable of being instantiated in an internal IT environment, if the tool is to be organisationally implemented [I5]. This requirement aligns with the broader context of the thesis, where RAM-related data may be security classified or export controlled, limiting the use of external cloud services.

Another topic discussed during the unstructured interviews concerned the selection of programming languages and development frameworks. It was pointed out that large organisations such as Saab typically have expertise across a wide range of technologies [I1, I5]. If the technology exists, someone will have the expertise in it [I1]. The choice of technology would primarily affect how easily the future team would take over the system development [I6]. Since the long-term technical owners of the system had not yet been determined, it was encouraged to select a suitable technology stack appropriate to develop the system within the time frame, while keeping maintainability in mind [I1, I6].

Preferences regarding certain system design approaches were expressed. In particular, the tool should not be developed directly within a spreadsheet-based environment, such as Microsoft Excel or Microsoft Access [I1, I2]. These options were considered unsuitable for use over a long period of time, as functionality may break due to updates to the Office software [I1]. However, since individuals who report failures may not require full access to the system itself, Excel spreadsheets were still considered useful as a part of the data collection workflow [I1, I2].

Finally, future integration capabilities of the system were voiced. Through further development after the thesis, the system should preferably be able to expose Application Programming Interface (API) endpoints to document management systems used within the organisation [I1].

3.3 Benchmarking

The benchmarking demonstration provided a concrete reference point for how a mature commercial reliability application is structured, namely as a suite of specialised tools rather than a single monolithic application. The portfolio overview clarified how different analysis needs are separated into dedicated products (e.g. Weibull analysis, system modelling, FMEA, and FRACAS), and that the software is complemented by supporting services such as training and technical support. This influenced the thesis scope by reinforcing the need to prioritise a well-defined core set of capabilities and to treat advanced analyses and broader tool-suite coverage as potential future extensions rather than immediate requirements.

A second key takeaway concerned interoperability and data exchange. The demo emphasised that practical adoption often depends on the ability to integrate with surrounding tools and workflows, either through direct interoperability within a tool ecosystem or through import/export mechanisms (including XML). This strengthened the requirement focus on export functionality and structured data outputs, and it also highlighted a potential future direction: enabling connections to business intelligence tools via database-backed reporting or standardised exports, provided that such integrations can be implemented within the constraints of the target environment.

The benchmarking, complemented by the provider's documentation, also provided concrete guidance on technical and mathematical aspects of reliability modelling. In particular, these sources describe how empirical failure data can be converted into cumulative failure probabilities for probability plotting. The vendor presents the use of Bernard's approximation for median ranks as a practical approach for assigning percentiles to ordered failure times, and describes it as a commonly used method in Weibull and reliability applications. This benchmarking input helped anchor the implementation choices in established industry practice and supported the selection of a transparent and repeatable method suitable for the expected data availability in the studied context.

Finally, the walk-through of their FRACAS-related functionality offered useful

insights into workflow design and usability for incident-driven reliability processes. The demonstrated approach centres on registering incidents using configurable templates, assigning incidents to problem reports, and then adding actions and repair information as the investigation progresses. Several interface elements illustrated how usability can support data quality, such as guidance for writing effective descriptions and the ability to select responsible parts from a hierarchical structure. At the same time, the demo underscored a recurring trade-off between completeness and usability. Extensive, highly detailed forms may increase analytical potential but can also create friction for contributors, especially when different roles are expected to add information over time. These observations informed the proposed requirements by motivating a structured yet lightweight reporting workflow, clear role-based responsibilities for data entry and review, and a deliberate prioritisation of essential fields and traceability over detailed configurability in the initial scope.

4 Requirements Engineering

This chapter presents the requirement analysis and specification that forms the basis for the digital tool. It describes how the requirements were engineered, resulting in the PRD provided in full in Appendix D.

4.1 Product Requirements Document

The PRD serves as the main specification document for the proposed digital RAM tool. Its purpose is to provide a structured basis for the system design by documenting the requirements in a consistent and traceable format. In the PRD, the requirements are organised through epics, user stories, functional requirements, and non-functional requirements. Epics represent high-level groupings of related functionalities or features. The functional requirements define the intended system behaviour in detail, while the non-functional requirements describe the qualities the system must exhibit in areas such as security, compliance, performance, usability, maintainability, and availability. These two requirement categories are further presented and discussed in the following sections.

To support prioritisation, each user story in the PRD was assigned a priority on a scale from 1 to 5, where the value reflects the relative urgency and importance of the functionality in relation to the purpose of the tool and the scope of the thesis. In addition to the requirements themselves, the PRD also includes supplementary material intended to clarify how the requirements should be interpreted in practice. This includes user interface concepts in the form of design wireframes, documentation of the Excel template used for problem report input together with the intended archiving procedure, definitions of action report data, and the design for S5000F-compliant XML export. The document also contains definitions of the RAM metrics and how they are calculated in the system. Together, these elements make the PRD more than a

list of requirements, as it also captures the intended operational context and provides guidance for the subsequent design and development work.

4.1.1 Functional Requirements

This subsection aims to describe the functional requirements of the digital RAM tool, including how they are structured, through epics, in the PRD and how they were derived from the requirements analysis. It also outlines the main considerations that guided their formulation, such as stakeholder needs, support in the literature, or benchmarking activities.

User Registration and Authentication

This epic covers the generic system functions related to how users access and interact with the system. It includes the handling of login and logout procedures, the registration of new users, and other fundamental authentication-related processes. Although these functions are standard in nature, they are essential for establishing a secure and accessible entry point to the application.

The overall rationale behind this epic was strongly guided by usability. The system should be intuitive for its users and designed in a way that makes the correct course of action clear, while minimising the risk of mistakes. In line with the needs expressed in the interviews, the authentication flow should therefore be straightforward and easy to understand, ensuring that even infrequent users can access the system without unnecessary friction.

System Configuration

This epic covers the administration of products and related system structures used for importing and analysing data. A central part of this epic is the ability for administrators to create and modify products, serialised units, breakdown items, and failure areas. This is essential for enabling the categorisation and multi-level analysis requested during the interviews. It also provides the foundation for access control by making it possible to manage user permissions on a product basis.

In addition to the product structures, an administrator should be able to configure categories that reliability data are supposed to be structured by, as expressed in the stakeholder interviews. Problem reports, if applicable, should be related to disciplines and failure areas, acting as sub-categories, depending on the nature of the affected product component. An example of discipline and subsequent failure area is Mechanical and Bearings. The system should allow administrators to manage failure areas within the disciplines.

User Management

This epic covers the administration of user accounts, roles, and product access within the system. Its purpose is to provide administrators with a clear and user-friendly interface of the system users and the management of them. This is important not only from an administrative perspective, but also for the overall usability of the system, as the interviews highlighted the need for intuitive and well-structured workflows. Ensuring that each user has access only to the functions and information relevant to their responsibilities is a key prerequisite for such usability.

The epic is also closely related to system security. The defined system roles, admin, viewer, reporter, and editor, are structured to ensure that users cannot access more information or functionality than required for their work tasks. For example, a reporter should not have access to analytical functions or the ability to create action reports, as these tasks fall outside that role's intended responsibilities. In this way, role-based access control supports both security and clarity in the user experience.

In addition to access management, this epic includes generic but necessary administrative functions, such as resetting passwords for users who have forgotten their credentials and disabling accounts that should no longer have access to the system. Since, e.g. password reset emails are not feasible in the intended secure IT environment, administrators must be able to assign a new password manually. Likewise, the ability to disable users is necessary to ensure continued control over system access when organisational roles or

responsibilities change.

Product Context Selection

This epic covers the selection of the active product within the system. The selected product determines which data and functionality are available to the user throughout the application. Upon login, the user selects a product (AUV62-AT or AUV62-MR), and this selection remains active until it is explicitly changed. As a result, all subsequent data handling, reporting, and analysis are carried out within the context of the selected product.

This design choice was based on the needs identified during the requirements analysis. No clear demand was expressed for comparisons between different products. Instead, the requested analyses were concerned with structures within a single product, such as units and breakdown items. For that reason, the system was designed around one active product context at a time. This approach simplifies navigation and reduces the risk of users working with the wrong dataset, while also supporting a clearer and more focused workflow.

Problem Report Import

This epic covers the functionality for importing problem reports into the system from structured datasets. Due to security constraints associated with deploying the tool in a secure IT environment, access to the application will be limited, and not all actors involved in reporting faults will be able to enter data directly into the system. For this reason, the reporting workflow is based on an external form that is completed outside the application and then uploaded by an authenticated user. In the proposed solution, this form takes the shape of an Excel file.

The Excel template is designed so that users can only enter the requested information and only in the intended format, for example through drop-down lists, numeric fields, and text fields. This is important both from a usability perspective, by guiding the user through the reporting process, and from a system perspective, by ensuring that the uploaded file can be interpreted correctly during import. This reasoning was further reinforced by the benchmarking,

where configurable incident templates and guided data entry illustrated how structured input support can improve both usability and data quality in incident-driven reliability workflows.

This epic also includes the upload and validation process itself. When a file is uploaded, the user should receive clear feedback on whether the report has been completed correctly, whether mandatory fields are filled out, and whether the entered values comply with the expected format. If the file does not meet these requirements, the user should be prompted to correct the Excel file and upload it again. The purpose of this workflow is to ensure that all imported data are reliable and accurately reflects the contents of the submitted report.

To support traceability and completeness, an approved and successfully uploaded Excel file should also automatically be renamed according to a system-generated report ID and then archived by the user in accordance with the defined procedure. In this way, the original submitted report is preserved and can be linked directly to the corresponding record in the system.

Problem Report Management

This epic covers the functionality for viewing and managing imported problem reports. Users with viewing access should be able to browse, search, filter, and inspect reports within the system. This is necessary to enable efficient oversight of the available information across different levels and categories, both to provide a structured overview of the dataset and to support the identification of issues that may require further attention.

Such functionality is also important for maintaining data quality. By being able to review the imported reports, users can identify potential errors or inconsistencies in the dataset, and perform adequate corrective actions, such as deleting the report. However, deletion in this context should not mean that the information is permanently erased from the system. Instead, reports should be marked as disabled and excluded from analyses while still remaining stored in the database. This approach supports traceability and makes it possible to restore records if something was removed by mistake, thereby contributing to

both data integrity and user-friendliness.

RAM Analysis

This epic covers the functionality for analysing collected failure data in order to support decision-making and structured RAM analysis. To address the analytical needs expressed by stakeholders in the interviews, while also reflecting the theoretical foundation established in the literature review, the system must support both detailed filtering of input data and a range of analysis metrics and visualisations.

Users should be able to filter the dataset based on relevant categories such as unit, breakdown item, discipline, and whether a report is classified as NFF, among others. This makes it possible to analyse specific subsets of the data and present them in a clear and accessible way, for example through pie charts showing distributions across categories and bar charts illustrating trends over time.

A further part of this epic concerns the calculation and presentation of key RAM-related metrics, including MTBF, MDBF, MTTR, and Availability. These measures should be available at different product levels depending on the applied filters, and they should be possible to view both as individual values and as graphical representations of average values over time. The user should also be able to compare observed values with previously entered predictions and targets, thereby supporting follow-up and assessment of system performance.

In addition to these more straightforward analyses, the epic also includes support for more advanced methods such as Duane and Weibull plots, together with their associated parameters and measures. For these analyses, users should be able to select the relevant basis for evaluation, such as time or distance, depending on the characteristics of the studied product.

Action Report Management

This epic covers the functionality for creating, managing, and viewing action reports within the system. The need for action reports emerged clearly in the

stakeholder interviews, where the ability to document and follow up corrective or preventive actions was identified as an important part of the overall workflow. To ensure consistency and completeness, the creation of action reports should be based on a structured set of predefined fields that guide the user in documenting the relevant information.

The intended users of this functionality are those with authority to act on analytical findings and initiate decisions. Since action reports are closely linked to the interpretation of problem reports and RAM-related analyses, they should only be available to users with the appropriate write permissions, namely admins and editors. This restriction supports both role clarity and controlled system usage.

To enable efficient handling of action reports over time, the system should support the creation, viewing, and removal of reports, as well as functions for searching, filtering, sorting, and browsing existing entries. This is necessary to provide users with a clear overview of ongoing and completed reports, and to ensure that action reports can be managed in a structured and practical manner.

Export Data

This epic covers the functionality for exporting selected information from the system in XML format, compliant with the S5000F in-service data feedback specification. Its purpose is to support interoperability with other internal systems within the organisation.

Based on the stakeholder interviews, the overall demand for export functionality from the RAM tool was limited. However, the need to export calculated RAM metrics and action reports for further use in other internal tools was identified. Since the receiving systems already apply S-Series-based data structures, and the organisation aims for broader alignment with the ASD S-Series suite of specifications, the ability to exchange information according to the suite was therefore considered an important functional requirement.

The choice of using specifically the S5000F specification as the basis for export was further supported by its role within the S-Series suite, as described in the

literature review. As it provides a structured format for exchanging in-service RAM-related feedback, its application matches the type of data intended to be exported.

The inclusion of export functionality was further reinforced by the benchmarking. During this activity, support for XML-based data exchange with surrounding systems was displayed. This confirmed that structured export is a common feature in reliability engineering tools and, even though the specific export needs varied between interviewed stakeholders, supports its inclusion as a functional requirement.

The export of calculated RAM metrics enables users to generate XML files containing MTBF, MDBF, MTTR, and Availability values based on the currently applied analysis filters. This way, the exported values are consistent with the context presented in the user interface and can be further used in other tools.

The export of action reports enables users to transfer documented decision-support information related to corrective actions, such as product design changes, updates of technical publications, and educational efforts, to other systems. Since action reports represent structured documentation of analysis observations and their associated actions, exporting them provides other systems with the context for change requests and decision-making.

Authorisation Warnings

This epic covers the functionality for handling and displaying user feedback when access to the system or specific functionality is restricted due to missing roles, product selection, or insufficient authorisation. Its purpose is to provide clear guidance to users and to prevent unintended system usage by making access limitations visible and understandable.

To support this, the system should display warning messages when users attempt to access parts of the application for which they are not authorised, when no product has been selected, or when no relevant role has been assigned. Such feedback improves usability by helping users understand why an action

cannot be performed, hence reducing the risk of confusion.

4.1.2 Non-Functional Requirements

In addition to the functional requirements, the PRD defines a set of non-functional requirements that describe the qualities the system must exhibit in use and in future development. These requirements are grouped into six categories: security, compliance, performance, usability, maintainability, and availability. More specifically, they state that the system must handle authentication securely, support S5000F-compliant XML export, provide responsive performance in both interaction and analysis views, and offer a consistent and intuitive user experience with clear feedback and validation.

The non-functional requirements also emphasise that the system should be modular and structured in a way that facilitates future extension, for example through the addition of additional RAM metrics, and that it should handle unexpected errors smoothly. Taken together, these requirements complement the functional scope by ensuring that the tool is not only capable of performing the intended tasks, but is also suitable for secure, reliable, and maintainable use in its intended organisational context.

5 Development Process

This chapter describes how the requirements identified in the previous chapter were translated into a working prototype of the digital RAM tool. It presents the overall development strategy, the architectural decisions, and the handling of challenges such as structured RAM data input, S5000F-compliant export generation, and iterative refinement of requirements during development. The chapter is meant to explain the process of how the system was developed, rather than provide an in-depth technical documentation of the system.

5.1 Development Strategy

As concluded from the stakeholder interviews, the intended system environment has significant security constraints, which limit the feasibility of adopting commercial tools. Hence, the actual development of the software was an important part of this thesis. Even though the objectives of the thesis possibly could have been sufficiently addressed through the presented requirements engineering work and the associated PRD, the development work provided additional insights. Developing a functioning system made it possible to assess the feasibility of the identified requirements and to validate them together with future system users.

The choice to develop the tool internally was also supported by the need for long-term system control. The interviews highlighted that current external tools, in similar areas, have limits in both responsiveness to requested changes and the ability to resolve defects. Having the tool internally developed makes it possible for future functionality to remain aligned with company-specific workflows and evolving product support needs.

The development was carried out in parallel with the requirements engineering process described in the previous chapter. Rather than treating requirements specification and development as strictly sequential activities, the work fol-

lowed an iterative approach in which architectural decisions and functionality were progressively aligned with the evolving PRD. This made it possible to continuously evaluate whether the identified requirements were implementable, and to refine both requirements and design decisions where necessary.

The produced digital tool demonstrated how the requirements captured in the PRD can be translated into a coherent technical solution. In this way, the prototype supports both evaluation of the proposed system design and continued development of the tool beyond the scope of the thesis.

5.2 System Architecture

The system architecture of the prototype was designed to support the structured handling, analysis, and exchange of RAM-related data as defined in the PRD. In particular, the architecture supports hierarchical product structures, role-based user access, structured problem and action report handling, filtering-based RAM analysis, and generation of S5000F-compliant export messages.

To support these objectives, the web-based application was developed using a layered architecture separating the user interface (front-end), application logic (back-end), and data storage (database). This separation was considered important both for maintainability and continued development of the tool beyond the scope of the thesis.

5.2.1 Front-End Architecture

The front-end was designed to support user-friendly interaction with RAM data while reflecting the identified workflows. It was structured to clearly guide users through tasks such as selecting the product context, importing problem reports, performing analyses, and managing action reports.

A central element of the user interface concerns the persistent product context selection. Since all analyses and data handling activities are performed within the scope of a selected product, it was important to present the active product selection clearly throughout the application. This reduces the risk that a user

performs operations on an unintended product.

Role-based interaction formed another important aspect of the front-end design. As different user roles are intended to perform different tasks within the system, the interface adapts available functionality depending on the assigned role. This supports usability by ensuring that users are presented only with the actions relevant to their responsibilities.

The front-end also supports filtering-based interaction with analysis data. Since RAM analyses are performed across product structures, customers, failure areas, etc., the interface was designed to allow users to create queries through a selection of relevant parameters before viewing calculated metrics or visualisations. Hence, the front-end supports exploratory analysis workflows through a mutual, user-friendly filter component.

5.2.2 Back-End Architecture

The back-end architecture was designed to support structured processing of RAM data while ensuring consistent enforcement of access control, problem report validation, filtering logic, and export functionality across the system. In contrast to the front-end, which primarily supports user interaction, the back-end is responsible for implementing the application logic.

A primary objective of the back-end was to ensure consistent handling of hierarchical product structures. Since RAM analyses may be performed at product, unit, or breakdown-item level depending on the selected analysis scope, back-end services were structured to support retrieval and aggregation of data across these levels. This made it possible to calculate and present RAM metrics in a way that reflects the relationships between system elements, rather than treating reports as isolated records.

Another responsibility of the back-end concerns the role-based access control. Although the front-end presents available functionality depending on assigned role, access control is enforced by the back-end. This ensures secure data retrieval and modification regardless of available front-end functionality.

The back-end also includes dedicated functionality for generating S5000F-compliant XML export messages. Since the generated messages are intended to be structurally validated by the receiver, this capability was implemented independently of the presentation layer. This ensures that export messages reflect the selected analysis scope and stored RAM data in a consistent and traceable manner.

5.2.3 Database Design Considerations

The database was designed to support structured storage and traceable relationships between configuration, user, report, and analysis data. Since the system is intended to manage data across multiple structural levels and time periods, the database needed to support both hierarchical representation of entities and consistent relationships between them.

An important data schema decision concerned how to represent the hierarchical product structures for problem report validation and relation. Through the stakeholder interviews, after several iterations, it was concluded that a product must be able to be referenced by units and breakdown items. The breakdown items act as sub-divisions of the product, all the way down to component level. To achieve this structure, each breakdown item, except for the top-level items, reference its parent breakdown item and associated product. This way, aggregation of RAM-related data and filtering of analysis results is enabled.

Data quality, for accurate analyses, was highlighted as essential for the tool during the interviews and literature review. This meant that, e.g. inaccurate problem reports needed to be handled in a way that they did not affect the metric calculations or visualisations. However, since almost all application data are highly coupled, simply deleting them is not optimal. For example, deleting a unit which is not in operation any more would result in all its problem reports not having a unit reference. Rather than permanently removing records when deleted, these remain stored in the database and are excluded from calculations through controlled status handling. This approach supports traceability and accurate analyses, while reducing the risk of unintended information loss.

5.3 Handling of Structured RAM Data

Since the RAM analysis directly depends on the quality and consistency of the underlying dataset, controlled import, validation, and storage of problem reports was essential. Concluded from the interviews, not all actors involved in reporting RAM data are expected to have access to the application. For this reason, the system was designed to support structured import of externally filled out problem reports using a predefined Excel template. This approach enables controlled data entry outside the application while still ensuring that imported information can be interpreted consistently within the system.

The Excel template guides users in entering information using predefined field structures such as drop-down selections, numeric values, and descriptive text fields. This ensures that the submitted information follows the expected format and reduces the risk of inconsistencies during import to the application. When a report is uploaded to the system, its contents are interpreted according to the template structure and mapped to the corresponding internal data fields.

Validation logic was incorporated to ensure that submitted reports comply with the expected data structure before they are accepted for storage and analysis. These validation mechanisms include verification of conditionally required fields, checking that entered values follow the expected format, and confirmation that referenced system entities correspond to existing application configuration.

For improved user experience, as requested per the interviews, feedback to users during the import process is provided. When inconsistencies or missing values are detected, the system indicates which corrections are required before the report can be accepted. Once successfully imported, the problem report becomes part of the structured dataset used for RAM analyses.

5.4 Implementation of S5000F-Compliant Data Export

Implementing export functionality according to S5000F required interpretation and tailoring of the specification, in order to identify which use cases, UoFs, and

data classes were relevant for representing the selected information types. Since the specification is designed to support a wide range of feedback scenarios, only a subset of the available structures was required for the intended application. The tailoring process therefore focused on identifying a consistent mapping between internally stored data and the corresponding S5000F data classes.

5.4.1 Tailoring Strategy

The tailoring of the specification was performed through a combination of top-down interpretation of its hierarchy and bottom-up inclusion of required data elements within the schema. The bottom-up approach slightly neglects the tailoring method as presented in the specification and literature review, but for some mappings it proved highly efficient.

The top-down part of the tailoring process followed the structure of the specification, beginning with identification of suitable use cases for representing the exports. From these use cases, the corresponding units of functionality were analysed in order to determine which data classes were required to represent calculated RAM metrics and action report information within a consistent XML message structure.

Complementing this approach, a bottom-up schema exploration strategy was used to identify appropriate data classes for representing individual data values. For example, specific metric identifiers such as MTBF were located within the schema and traced to their corresponding data class in order to ensure that exported values were placed within valid S5000F message contexts.

Together, these complementary strategies supported generation of export message templates aligned with both the formal structure of the specification and the practical data representation needs of the digital tool. To ensure the produced message templates were compliant with the specification, a script validated them against the S5000F XSDs.

5.4.2 RAM Metrics Export Messages

Export messages containing calculated RAM metrics were generated using the S5000F use case *UC50304: Report Availability*. Since calculated RAM metrics may be generated at different product-structural levels depending on the selected analysis scope, separate export templates were defined for product-, unit-, and breakdown-item-level contexts. In the remainder of this thesis, these analysis scopes are referred to as segments.

This segmentation reflects how RAM analysis is presented in the application, depending on filters set by the user. By preserving this structural context in the exported messages, the calculated values can be interpreted consistently by the receiving system.

Each exported metric is represented by the S5000F class *reportable metric*, which provides a standardised way of identifying the type of performance value. Associating each calculated value with its type and product-level enables the receiving systems to interpret the exported information correctly, without requiring additional contextual information outside the generated message.

To ensure S5000F-compliant message generation, even when calculated values are not available (e.g. due to insufficient data), message templates were produced representing both complete and incomplete metric sets.

An important remark is that MDBF is not included among the valid values of the S5000F schema. This is not an issue according to tailoring of the specification, since valid values are allowed to be added, as described in the literature review. However, this means that the receiver of exported RAM metrics messages must add MDBF to the *performanceParameterTypeCode* for these messages to pass validation.

5.4.3 Action Report Export Messages

Export messages containing action report information were generated using the S5000F use case *UC50302: Influencing future design*. This export message had a clearer document-based origin, compared to the previous export, which

simplified the top-down tailoring process. However, not all values had directly corresponding mappings, such as decision maker role and change reference. For these instances the *remark* class, which is included in all S5000F data classes, was used to provide the necessary context.

This export message generation also used a template representing cases where optional action report information fields were not available. This ensured that exported messages remain compliant, while allowing flexibility in how report information is recorded within the application.

5.5 Iterative Development and Requirement Baselines

During the early stages of development, a set of core system capabilities was established as requirement baselines in the PRD in order to enable continued development work while the remaining requirements were still being refined. These core functionalities included user authentication, role-based access control, and application configuration, which were considered necessary for supporting all subsequent functionality. Establishing these elements early made it possible to continue development without becoming overly dependent on stakeholder response times associated with refinement of more specialised functionality.

Other parts of the system, such as the detailed structure of export functionality, analytical features, and representation of product structures, were instead refined progressively during the development work. Through this approach, tool development and PRD refinement were carried out in parallel. This allowed development decisions and requirements interpretation to inform each other while maintaining alignment between the PRD and the developed application.

The initial plan was to have the supervisors incrementally evaluate and test the tool as it was developed. This was meant to be achieved through setting up a cloud-hosted instance of the application, which would be easily accessible for the supervisors. However, this approach was not possible, since a cost center for the hosting arrangement could not be established during the thesis work. Instead, review and verification of developed functionality were carried out

through on-site formal reviews, where the application was hosted locally.

Although this meant that validation was not performed incrementally throughout the full development process, the completed functionality could still be reviewed against the PRD. The developed tool was assessed to satisfy the intended scope of the thesis, while the limitations of the testing setup remained primarily logistical rather than methodological.

5.6 High-Level Design Documentation

In addition to the development described in this chapter, a separate HLDD was produced to support continued development of the RAM tool beyond the scope of the thesis. The purpose of this document is to provide a structured technical description of the system architecture, as a reference for future developers within the organisation.

Since the primary audience of the HLDD consists of developers responsible for maintaining and extending the system after completion of this thesis, the document is written at a higher technical level than the main body of the thesis. For this reason, the HLDD is not included in the report, but is instead treated as a separate handover document intended to support future maintenance and further development of the system.

6 The Developed System

This chapter presents the developed digital RAM tool together with the supporting Excel-based data collection template used for problem reporting. The purpose is to illustrate how the overall workflow described in this thesis is realised through the developed tool, from structured data capture to import, management, and analysis in the tool. The workflow includes collecting and structuring problem report data in a consistent format, importing and managing problem reports, performing RAM-oriented analyses using configurable filters, documenting follow-up actions, and maintaining the configuration data needed for consistent reporting and analysis.

6.1 The Digital Tool

This section presents the main user interfaces of the developed digital RAM tool. The screenshots in Figures 6.1–6.16 provide an overview of the implemented functionality and the intended user interaction.

6.1.1 Landing Page

After the user has signed in (i.e. is authenticated), the tool presents a simple landing page where the user may select a product to work with. This page is shown in Figure 6.1. The options available in the globally present top navigation bar depend on the user's role. In this case, the user is an admin and therefore has access to all pages in the tool.

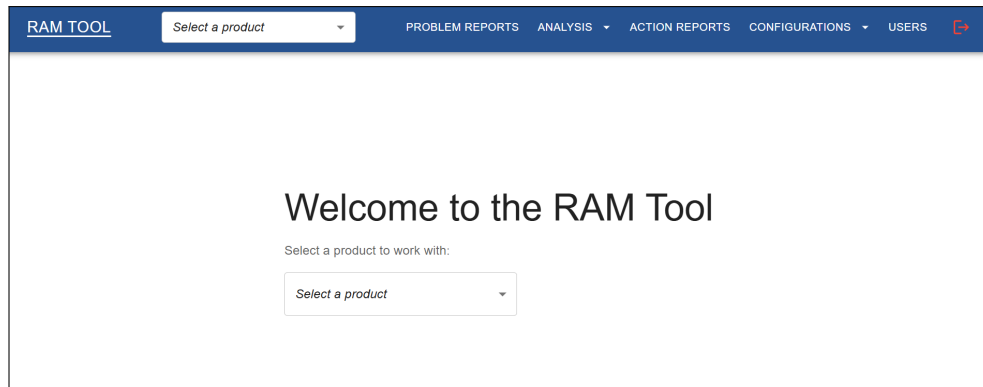


Figure 6.1: Landing page of the RAM tool, where the user selects a product to work with.

6.1.2 Problem Reports

The problem reports view, shown in Figure 6.2, provides the main interface for browsing imported problem reports. The page is built around a searchable and filterable list, allowing users to narrow down reports by Report ID, Unit ID, date, Breakdown Item, Discipline, Failure Area, and Root Cause. In addition to browsing problem reports, the view includes the ability to import new problem reports. The list also provides navigation to a detailed problem report view, allowing a user to inspect a specific report.

RAM TOOL

AUV62-MR

PROBLEM REPORTS

ANALYSIS

ACTION REPORTS

CONFIGURATIONS

USERS

Problem Reports

IMPORT NEW PROBLEM REPORT

ALLACTIVEDISABLED

Report ID	Unit ID	Failure Date	Breakdown Item	Discipline	Failure Area	Root Cause
Search...	Search...	From To	Search...	All	Search...	All
PR-ANA001	SA-001	2025-05-10				
PR-ANA002	SA-001	2025-05-25	PA - Thruster Motor	Mechanical	Motors	Component error
PR-ANA003	SA-002	2025-06-05	NB - GPS Receiver	Electrical/Electronic	Circuit Boards	Component error
PR-ANA004	SA-001	2025-06-15				
PR-ANA005	SA-003	2025-06-20	EA - Battery Pack	Electrical/Electronic	Battery Systems	Component error

Figure 6.2: Problem report list view with search and filter controls, and functionality for importing new problem reports into the dataset of the selected product.

6.1.3 Analysis

Once problem reports are imported, the Analysis section provides multiple perspectives on the same underlying dataset.

Failure Distributions

Figure 6.3 provides an overview of the Failure Distributions page, which summarises a filtered set of problem reports through distribution charts, Figure 6.4, and a trend plot showing report volume over time, Figure 6.5. The filter, Figure 6.6, is mutual for all applicable analysis pages, allowing the user to customise their analysis. This view is primarily intended as an exploratory overview that helps users understand the composition of the data before moving into more calculation-driven analysis.

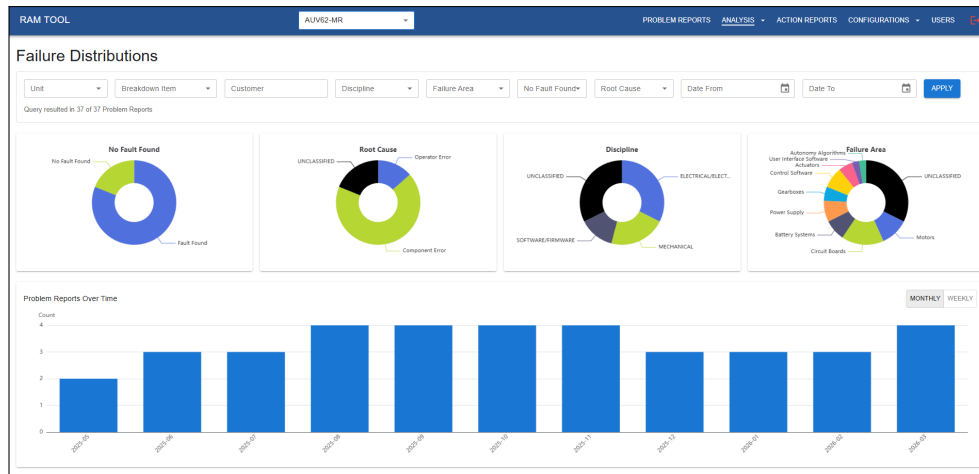


Figure 6.3: Failure Distributions view summarising the filtered problem report dataset using distribution charts and a report-volume trend over time.

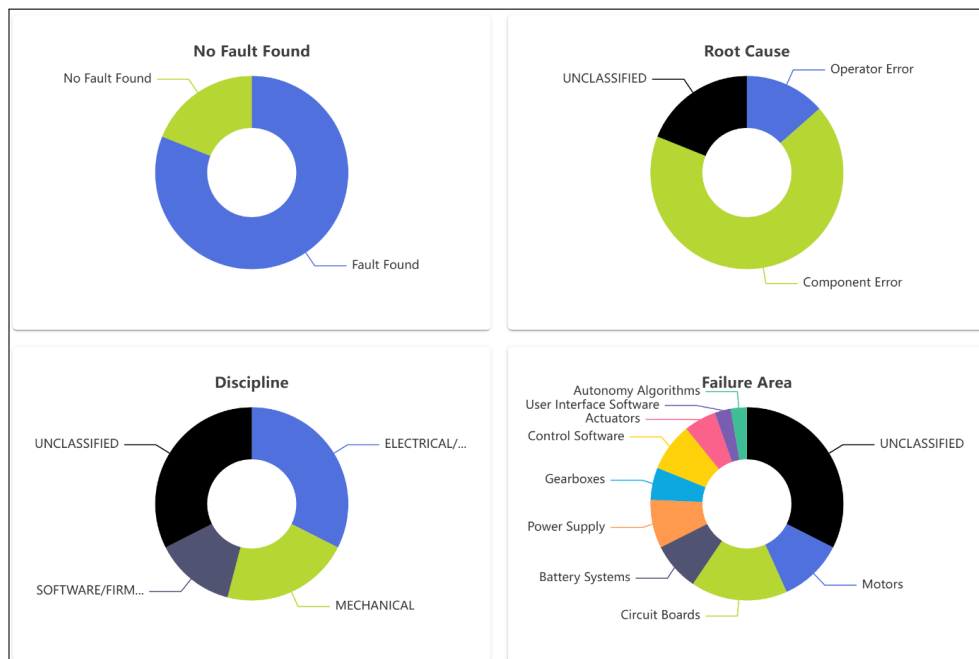


Figure 6.4: Distribution charts in the Failure Distributions view, summarising the filtered problem reports by key classifications (NFF, root cause, discipline, and failure area).

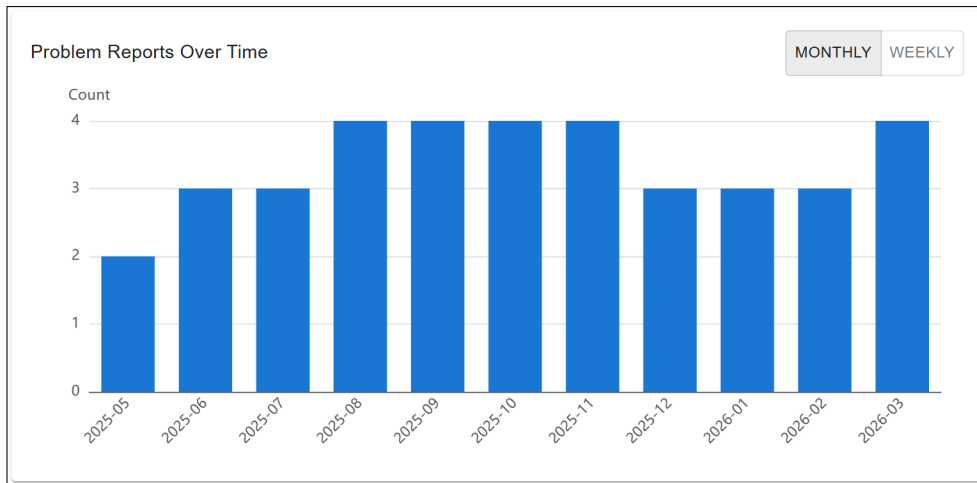


Figure 6.5: Bar chart of problem report volume in the Failure Distributions view, showing the number of reports in the filtered dataset aggregated by month or week.

Figure 6.6: Mutual filter component for applicable analysis views, used to define the subset of problem reports included in the analysis.

RAM Metrics

The next page within the Analysis section is the RAM Metrics view, of which Figure 6.7 provides an overview. The view includes calculated RAM metrics for the selected segment(s) (product, units or breakdown items) based on the applied filters, shown in Figure 6.8. The page supports segmentation on different levels and compares calculated RAM metrics with predicted and targeted values where applicable. A cumulative trend plot is also provided, Figure 6.9, to show how metrics evolve as additional operational data are accumulated over time. The view furthermore supports exporting calculated values in an S5000F-compliant XML format.

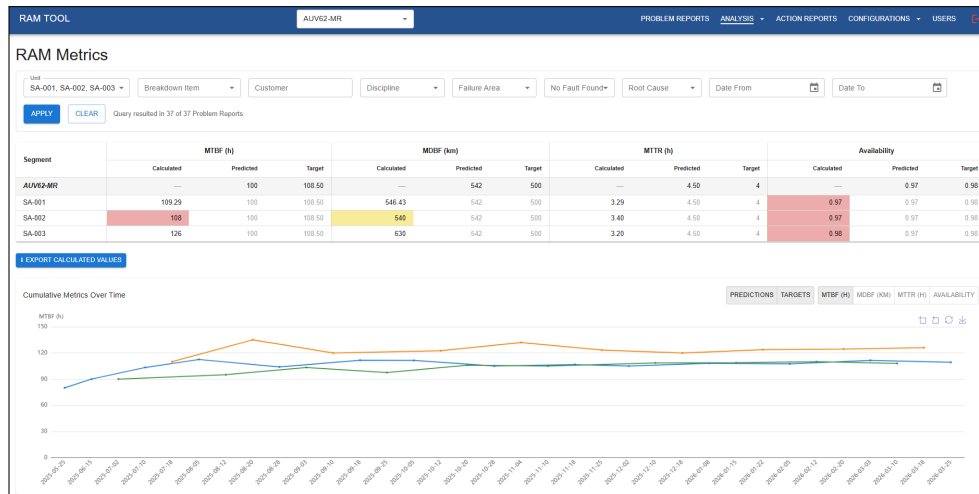


Figure 6.7: RAM Metrics view showing calculated metrics per selected segment(s), together with predicted and targeted values, and a cumulative metrics trend over time.

Segment	MTBF (h)			MDBF (km)			MTTR (h)			Availability		
	Calculated	Predicted	Target	Calculated	Predicted	Target	Calculated	Predicted	Target	Calculated	Predicted	Target
AUV62-MR	—	100	108.50	—	542	500	—	4.50	4	—	0.97	0.98
SA-001	109.29	100	108.50	546.43	542	500	3.29	4.50	4	0.97	0.97	0.98
SA-002	108	100	108.50	540	542	500	3.40	4.50	4	0.97	0.97	0.98
SA-003	126	100	108.50	630	542	500	3.20	4.50	4	0.98	0.97	0.98
SA-004	—	100	108.50	—	542	500	—	4.50	4	—	0.97	0.98

EXPORT CALCULATED VALUES

Figure 6.8: RAM Metrics view showing calculated RAM metrics for the selected segment(s), with comparisons to predicted and targeted values.

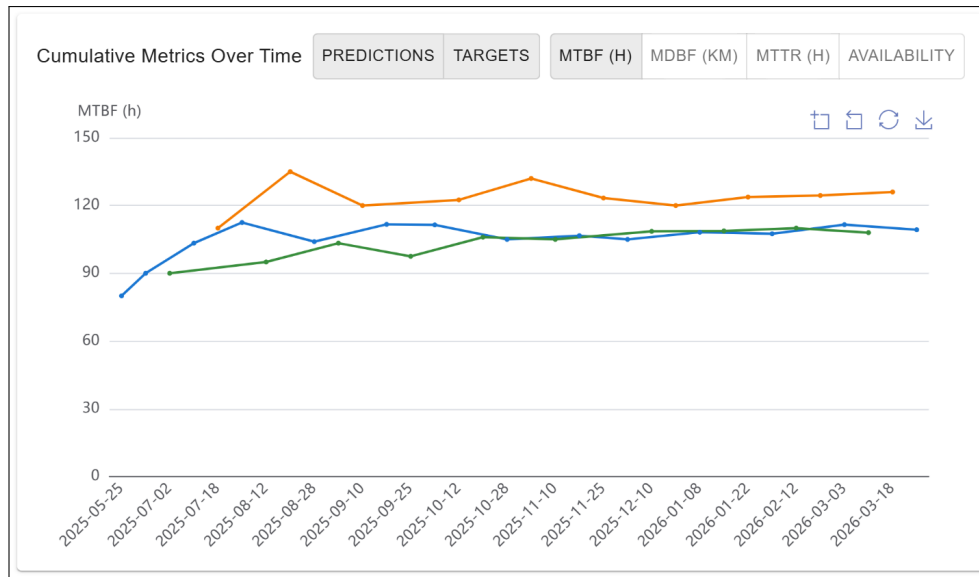


Figure 6.9: Plot of cumulative RAM metrics over time in the RAM Metrics view, illustrating how selected metrics evolve as additional operational data are accumulated.

Reliability Trends

Figure 6.10 provides an overview of the Reliability Trends view, which is dedicated to reliability modelling and trend analysis. The page provides a Duane Plot, Figure 6.11, and a Weibull plot, Figure 6.12, as well as their estimated parameters. This allows the user to assess trends and distributional behaviour for the selected segment(s). The intention is to support both comparative analysis between segments and interpretation of model parameters in a way that is consistent with the calculations described in the thesis.

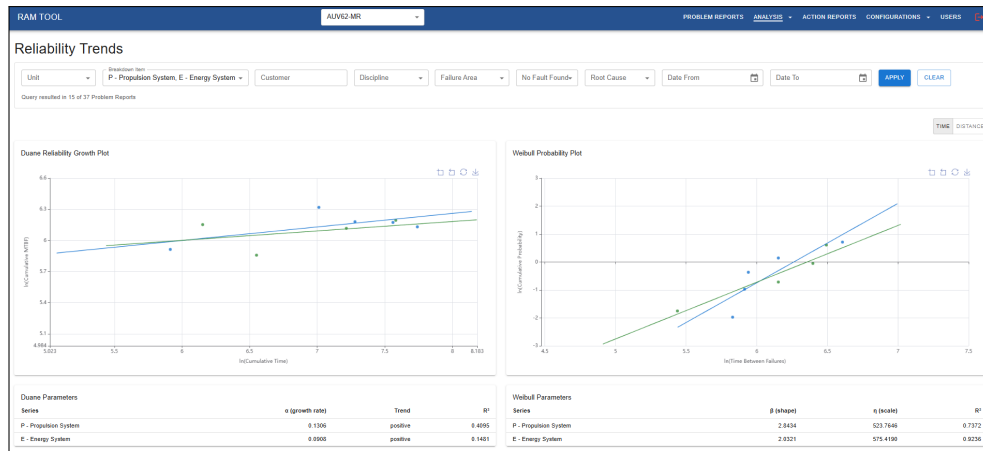


Figure 6.10: Reliability Trends view providing Duane- and Weibull-based analysis for selected segment(s).

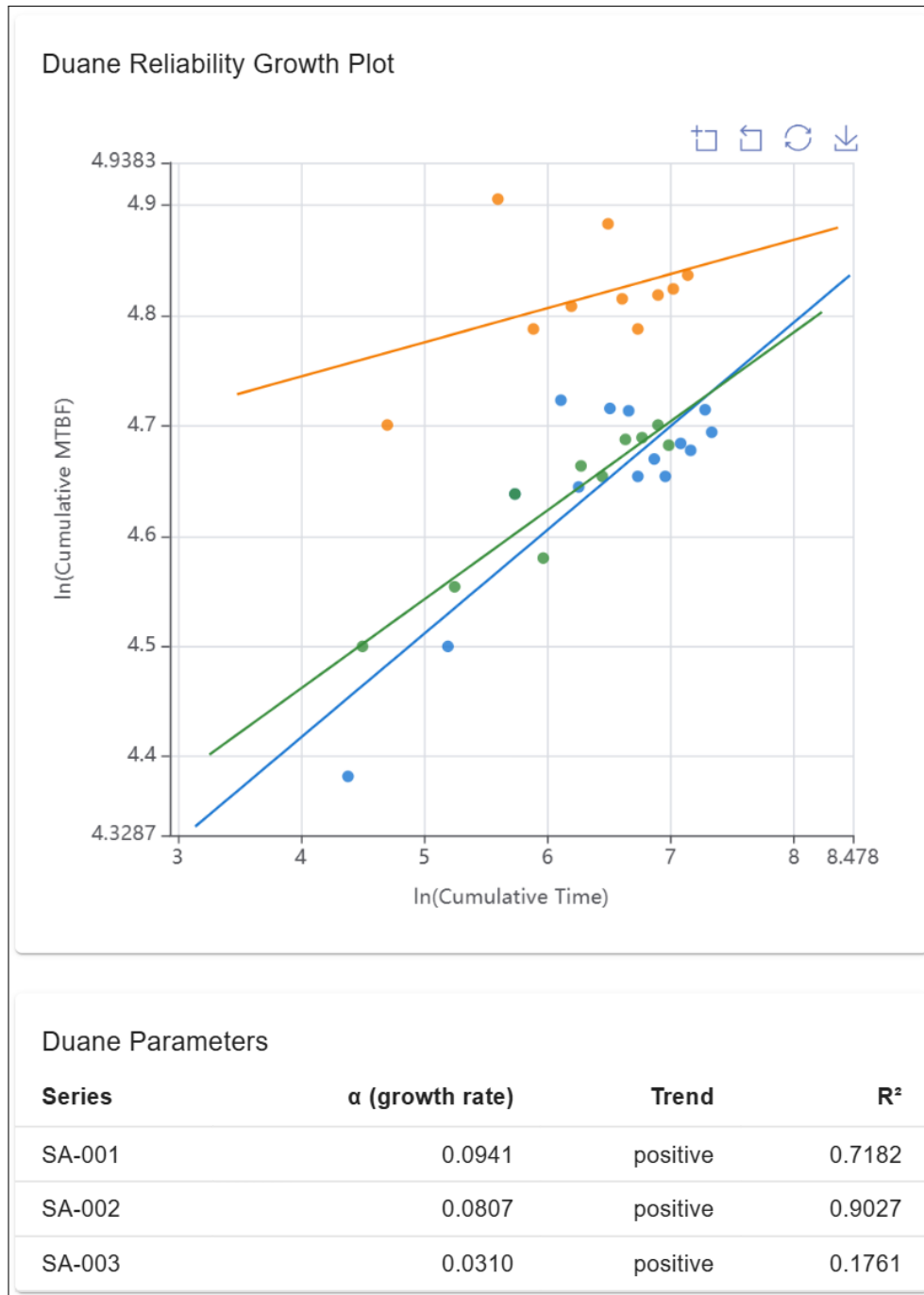


Figure 6.11: Duane reliability growth plot in the Reliability Trends view, illustrating reliability growth over accumulated operating time for the selected segment(s), together with the estimated growth rate (α) and the trend direction (positive/negative) with goodness-of-fit (R^2).

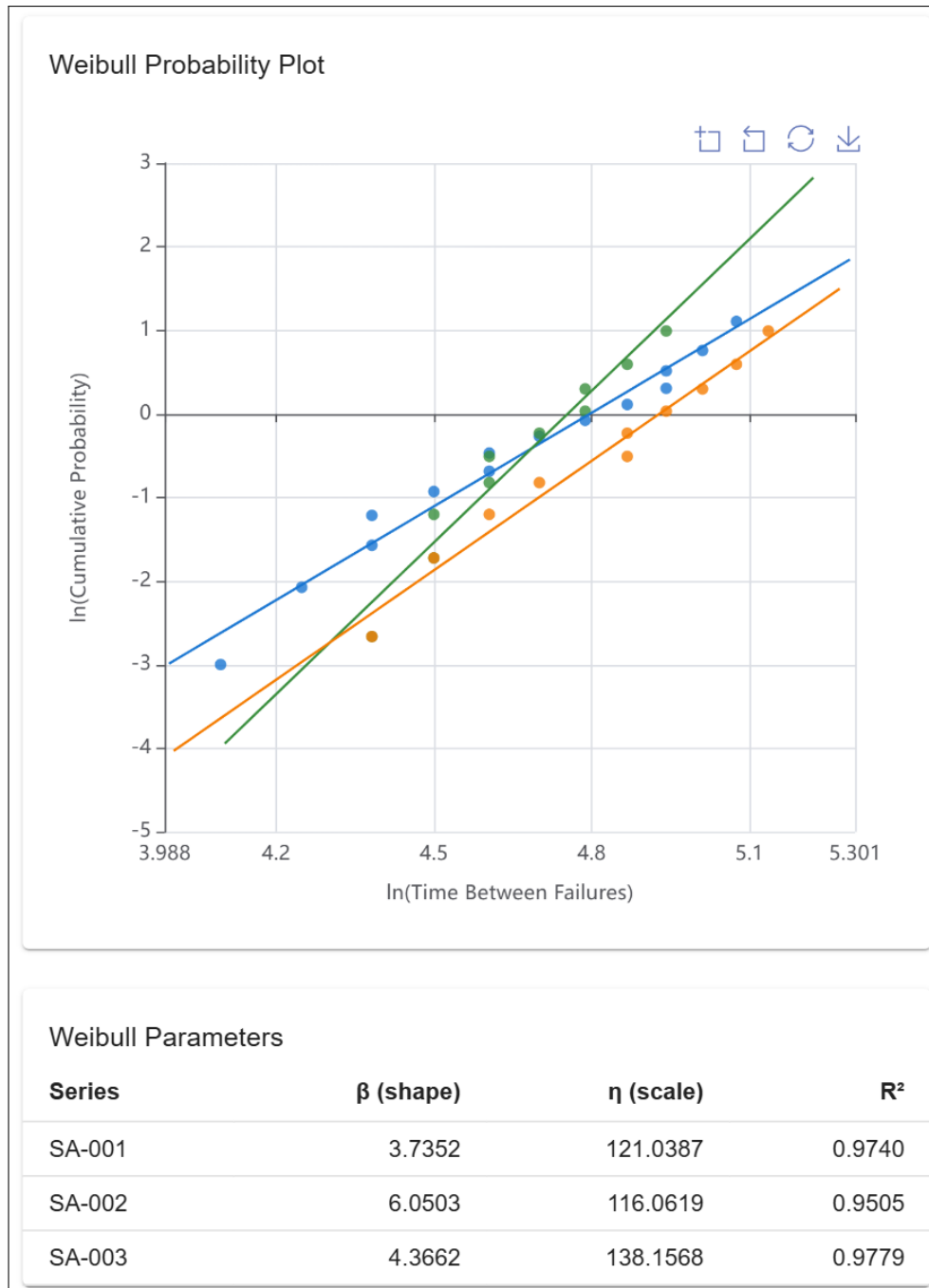


Figure 6.12: Weibull probability plot in the Reliability Trends view, showing the fitted two-parameter Weibull model for the selected segment(s) and the corresponding estimated parameters (β and η) with goodness-of-fit (R^2).

6.1.4 Action Reports

To document decisions and create traceability from analysis to follow-up, the tool includes the possibility to create and export Action Reports. Figure 6.13 shows the action report list, where users can create, edit, browse, and filter action reports. Each entry captures key information such as decision maker, corrective actions, and references to change requests. This view is intended to support a structured improvement loop by making actions and decisions visible and searchable. The list provides navigation to a detailed report view, where a user may export the report in an S5000F-compliant XML format.

Report ID	Date ↓	Decision Maker	Corrective Actions	Change Reference	Created By	Updated By
AR-2026006	2026-04-02	Gustav			Oscar Cedell	Oscar Cedell
AR-2026005 ✓	2026-04-22	Jacob	Design Change, Technical Publications	MoM: 978987987	Oscar Cedell	Oscar Cedell
AR-2026004	2026-04-27	Oscar	Design Change, Educational Effort	CO: 123123123	Oscar Cedell	

Figure 6.13: Action Report list view used to document follow-up actions and decisions, including corrective actions and change references.

6.1.5 Configurations

In addition to the operational and analytical views, the tool contains configuration pages used to maintain product-specific reference data. Figure 6.14 shows the Product configuration view, where administrators define the product structure and associated entities required by the tool (e.g. units, breakdown items, and RAM metric targets and predictions). These configurations form the basis for consistent classification, segmentation, and filtering throughout the rest of the application.

Product

CLOSE

Name

AUV62-MR

Active

Yes

SAVE

RAM Metrics ?

Predictions

MTBF (h)

100

MDBF (km)

542

MTTR (h)

4,5

Availability

0,97

Targets

MTBF (h)

108,5

MDBF (km)

500

MTTR (h)

4

Availability

0,98

Units

Unit serial number...

ADD

Name ↓

Search

ALL

ACTIVE

DISABLED

SA-001

SA-002

SA-003

SA-004

Top-Level Breakdown Items

LCN...

Name...

ADD

Lcn ↓

Name

Search

ALL

ACTIVE

DISABLED

C

Communication System

E

Energy System

N

Navigation System

P

Propulsion System

Figure 6.14: Product configuration view used to maintain product-specific reference data such as units, breakdown structure, and RAM metric targets.

Figure 6.15 illustrates the Failure Areas configuration view, where administra-

tors manage the Failure Area categorisation used during reporting and analysis. The view supports adding and disabling entries and groups Failure Areas by Discipline (i.e. Electrical/Electronic, Mechanical, or Software/Firmware). Maintaining this controlled structure is a prerequisite for organised reporting for reliable filtering and aggregation in the analysis views.

The screenshot shows the 'Failure Areas' configuration view in the 'RAM TOOL' application. The top navigation bar includes 'PROBLEM REPORTS', 'ANALYSIS', 'ACTION REPORTS', 'CONFIGURATIONS' (selected), and 'USERS'. The main content area is divided into three columns representing different disciplines: Electrical/Electronic, Mechanical, and Software/Firmware. Each column has a 'Failure area name...' input field with an 'ADD' button. Below each input field is a table listing failure areas. Each entry in the table has a 'DISABLE' button.

Discipline	Failure Area Name	Action
Electrical/Electronic	Battery Systems	DISABLE
	Circuit Boards	DISABLE
	Power Supply	DISABLE
Mechanical	Actuators	DISABLE
	Gearboxes	DISABLE
	Motors	DISABLE
Software/Firmware	Autonomy Algorithms	DISABLE
	Control Software	DISABLE
	User Interface Software	DISABLE

Figure 6.15: Failure Area configuration view used to maintain Discipline-specific Failure Area classification structure for structured reporting and filtering.

6.1.6 Users

Finally, access and responsibilities are governed by role-based access control. Figure 6.16 shows the Users management view, where administrators can manage user accounts and assign roles (e.g. reporter, analyst, or editor). This view is essential for implementing governance and ensuring that users have a sufficient level of access to the tool and its datasets. In addition, an administrator is allowed to assign configured products to users, further governing the user access control. A user is limited to data associated with their assigned products.

RAM TOOL AUV62-MR PROBLEM REPORTS ANALYSIS ACTION REPORTS CONFIGURATIONS USERS

Users

ALL ACTIVE DISABLED

Name ↓ Email Role

Search for a name... Search for an email... Select a role

Alice Andersson	alice.andersson@saabgroup.com	EDITOR
Jacob Persson	jacob.persson@saabgroup.com	ANALYST
Oscar Cedell	oscar.cedell@saabgroup.com	ADMIN

Figure 6.16: User management view for administering users and assigning role-based access within the tool.

6.2 The Excel Template

This section presents the Excel-based data collection template used to gather problem report data intended for import to the digital RAM tool. The figures show the three sheets included in the template: *Problem Report Form* (Figure 6.17, Figure 6.18, and Figure 6.19), *Revision History* (Figure 6.20), and *Reference Data* (Figure 6.21). Since the *Problem Report Form* exists in multiple variants depending on entered information, three variants of the sheet are included, followed by one example of *Revision History* and one example of *Reference Data*. The figures are provided to illustrate the structure, guided input fields (e.g. drop-down lists), and the intended reporting flow.

Product Information	
Product	
Unit ID	
Customer	

Failure Information	
Reporter Email	
Failure Date	
Run Time Before Failure	
Distance Before Failure	
Customer Return Date	
Labor Hours	
Failure Symptoms	
Troubleshooting Actions	

Root Cause Analysis	
No Faults Found	No
Root Cause	Operator Error
Operator Error Type	
Comments	

Figure 6.18: Excel template - Problem Report Form (Variant 2). Overview of the initial reporting sheet used for problem reporting, without NFF and with Operator Error.

Product Information	
Product	
Unit ID	
Customer	

Failure Information	
Reporter Email	
Failure Date	
Run Time Before Failure	
Distance Before Failure	
Customer Return Date	
Labor Hours	
Failure Symptoms	
Troubleshooting Actions	

Root Cause Analysis	
No Faults Found	No
Root Cause	Component Error
Component LCN	
Component Part Number	
Component Revision (PDM)	
Discipline	
Failure Area	
Failure Mode	
Corrective Actions	
Comments	

Figure 6.19: Excel template - Problem Report Form (Variant 3). Overview of the initial reporting sheet used for problem reporting, without NFF and with Component Error.

Revision	Date	Author	Change
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Figure 6.20: Excel template - Revision History. Complementary sheet used for revision history.

[illegible]

Figure 6.21: Excel template - Reference Data. Supporting sheet used to manage and add reference data for the template, such as values used in drop-down menus.

7 Discussion

This chapter reflects on the results in relation to the problem statement and the company context, focusing on practical considerations for deploying and adopting the proposed digital RAM tool at Saab Naval. It discusses expected challenges and opportunities during development, addresses limitations of the study, and outlines directions for future work.

7.1 Trade-Offs and Design Decisions

This section outlines the key trade-offs and design decisions made during the development of the digital tool, and explains the reasoning behind them.

7.1.1 Deriving Measures from Feasible Operational Data

The development of the digital RAM tool required several design choices where theoretically “optimal” solutions were weighed against feasibility in an operational context. In particular, the tool is intended to be used by personnel with limited time for reporting, and within a setting where access to certain data sources is constrained. The design therefore prioritises data that is practically obtainable with low reporting effort and high consistency, even if this means that some metrics are derived indirectly or that certain theoretically relevant measures are excluded.

A clear example is cycle-based reliability measures, such as Mean Cycles Between Failures (MCBF), which can be more appropriate than time-based metrics for some components in an AUV. Certain parts are used in discrete actuations rather than continuously, for example buttons, switches, or other operator interface elements. For such components, counting cycles would provide a more representative measure of exposure and ageing than hours of operation. However, within the studied context there is currently no practical or reliable way to collect cycle data across the relevant components without

introducing additional instrumentation or reporting burden. Since this would reduce feasibility and could lower data quality, cycle-based input was not included in the tool at this stage.

A related trade-off concerns how operating time and distance are captured for RAM calculations. In principle, the most informative input would be the time and distance since the last failure, as this is the exposure measure directly used in metrics such as MTBF and MDBF. Nevertheless, the tool was designed to collect time and distance since start of operation (i.e. accumulated runtime and accumulated distance) rather than “since last failure” values. This choice was made to reduce reporting effort and avoid manual calculations by operational personnel. There is infrastructure to read accumulated runtime since start, and distance can also be derived or estimated based on available operational data. Collecting exposure since the last failure would likely be possible, but it would require additional manual steps and could introduce additional input errors or inconsistencies. The tool therefore captures the most accessible and reliable inputs and derives the required “since last failure” measures internally, using the stored failure history and the accumulated values.

7.1.2 External Data Collection and Secure Import

A further design choice concerns how operational data are collected before it is imported into the tool. The data collection in this thesis is, as earlier mentioned, performed through an Excel template. The primary motivation for this choice is information security and the practical constraints of the target deployment environment. Since the tool is intended to be hosted within Saab Naval’s secure IT environment, external actors such as customers performing maintenance activities will not have direct access to the application. An external data collection mechanism is therefore required. Using Excel provides a pragmatic solution with low technical overhead and strong usability support, for example through drop-down lists, hidden fields, and explanatory text that guide the reporter and reduce input errors.

An alternative would be to develop a separate, standalone front-end for data

collection that outputs structured data according to S5000F, i.e. as an XML file. This would remove the dependency on Excel and align the data capture more directly with the standard. However, such an approach would require more development effort and stricter routines to remain secure. The collection interface must not store sensitive information, retain submitted data locally, or have any connection to the secure tool environment. In practice, the data collection front-end should be treated as having a similar risk profile as an unauthorised user gaining access to an empty Excel template: the web page itself must be non-sensitive, and no confidential information should persist outside approved environments. Under such a setup, the output would be an XML file that can be transferred to an authorised user for controlled import into the secure tool. This resembles common workflows from the benchmarking session where systems accepted structured XML imports, equivalent to how the tool developed in this thesis imports xlsx files. Overall, the Excel-based approach was selected as a feasible compromise between usability, effort, and security, while still enabling structured data capture that can be validated during import.

7.1.3 Failure Definition and Inclusion in Analysis

Another design choice concerns how failures are defined and how problem reports are included in the analysis. In the tool, all imported problem reports are treated as relevant failure events by default and are included in calculations, provided that they are not explicitly marked as disabled in the dataset. This means that NFF reports and events classified as operator errors are, in principle, handled in the same way as other incidents. The rationale is to preserve the full operational feedback picture and avoid prematurely excluding events that may still carry analytical value, for example by indicating intermittent faults, diagnostic limitations, training needs, or interface issues.

At the same time, the tool accounts for the fact that different analyses require different definitions of what should count as a failure. If the user aims to perform a more strict failure analysis focused on technical component failures

only, it is straightforward to exclude NFF cases and/or operator errors. The tool therefore supports filtering based on problem classifications, allowing users to adapt the dataset to their specific analytical purpose. This approach both keeps the default analysis complete and transparent, while enabling narrower interpretations of failure behaviour when required by the user.

7.1.4 Availability Metric: MTTR versus MDT

Another design choice concerns how availability is calculated and which repair-related metric is used. In the literature review, availability was defined and discussed using MDT, which is based on the total time an item is unavailable, i.e. its downtime. In the developed tool, availability is instead calculated using MTTR, which accounts only for the time spent repairing the item. This choice was made in consultation with stakeholders at Saab Naval and reflects what was considered most decision-relevant in the studied context.

The reason for this is that MDT is strongly influenced by factors that extend beyond the technical repair activity itself, such as missing spare parts, transport and logistics lead times, administrative delays, or support agreements that have expired. While these factors are important for operational availability, they can obscure the signal that stakeholders primarily wanted to follow up on, that is, the maintenance and repair effort required when failures occur. MTTR provides a more direct indication of the time spent repairing and is therefore more useful for understanding maintainability and for identifying improvement opportunities that Saab Naval can more directly influence. Consequently, the tool uses MTTR-based availability, while recognising that a MDT-based availability measure could be added in future versions if desired.

7.1.5 Weibull Modelling Choices

A further design choice concerns the Weibull model formulation used for parameter estimation and probability calculations. The tool applies a two-parameter Weibull model, defined by the shape parameter β and the scale parameter η , rather than a three-parameter Weibull model that also includes

a location parameter γ . A three-parameter Weibull can be useful when there is evidence of a failure-free, non-zero time period. However, reliably estimating γ typically requires larger datasets and high confidence in the representativeness of early-life data, since the location parameter can otherwise become unstable and overly influence the fit. In the studied setting, failure datasets are expected to be limited for many components and may reflect heterogeneous operating conditions and maintenance practices. The two-parameter model was therefore chosen as a pragmatic compromise as it is widely accepted, easier to communicate, and tends to provide more stable estimates under limited data availability. If future deployments yield larger and more consistent datasets, support for a three-parameter model could be reconsidered for components where clear threshold behaviour is observed.

Within the selected model, another design choice concerns how failure probabilities are estimated when producing Weibull plots. In the tool, cumulative failure probabilities are assigned to the ordered failure times using Bernard's approximation of the median ranks. This option was selected because it is straightforward to compute and, as mentioned earlier, a commonly used method in the industry. An alternative would have been to use other plotting-position formulas (e.g. based on $\frac{i}{n}$ or $\frac{i-0.5}{n}$), which place the probability points slightly differently. Another alternative could be to avoid plotting positions altogether and instead estimate Weibull parameters directly using maximum likelihood estimation (MLE). While MLE can be preferable when sufficient data are available and assumptions are well supported, it is also more sensitive to model assumptions and requires careful handling of edge cases. Given the expected data availability and the goal of keeping the method transparent and repeatable in an operational tool, Bernard's approximation was considered an appropriate compromise between theoretical rigour and practicality.

Right-censored data, referring to observations where the failure time is not observed within the examined period, is a relevant consideration in reliability modelling for the studied context. In operational settings, many components will have accumulated operating time or distance without having failed at the time of analysis. Such observations contain information about survival,

and reliability methods such as Weibull parameter estimation can explicitly incorporate them. In the current version of the tool, where censored observations are not modelled, the implemented Weibull-based calculations and plots are based on observed failure events only. This choice was made as a pragmatic simplification. Explicit censoring support requires consistent exposure data for non-failed items, including clear definitions of the population at risk and reliable tracking of operating time or distance for components that remain in operation. In the studied product context, such information is not available at the necessary level of detail without additional data collection and governance routines. Introducing censoring without robust exposure data could therefore create a false sense of accuracy. Consequently, the tool prioritises transparent calculations based on the available failure history, while recognising that the lack of censoring may bias estimates, particularly when failures are rare or datasets are small. Supporting right-censoring is a suitable extension for future versions if consistent exposure tracking can be established.

7.2 Limitations

This section outlines the main limitations of the study and explains how they influence the interpretation of the results. The limitations primarily relate to the scope of the developed digital tool, which was not deployed and evaluated in live operations, the context-specific nature of the study, and the reliance on interviews and a literature review. Making these limitations explicit helps frame the contributions of the thesis and provides context for the conclusions drawn from the work.

One limitation concerns the scope of the work in relation to organisational implementation. This thesis delivers a functional digital RAM tool and a proposed workflow, but it does not include deployment in Saab Naval's secure environment or a concrete organisational plan for adoption, training, and long-term follow-up. As a result, the study cannot draw firm conclusions about sustained use, long-term data quality, or realised effects on decision-making and improvement actions in daily operations.

A second limitation is the context-specific nature of the study. The work was conducted within Saab Naval and focused on two specific products (AUV62-AT and AUV62-MR). Organisational processes, culture, information security constraints, and the surrounding system landscape may differ across projects, departments, and companies. This limits the extent to which the findings and the proposed solution can be directly generalised without adaptation.

Another limitation is that some key findings are based on the conducted stakeholder interviews, which introduces risks related to subjectivity and bias. Interview responses may reflect the perspectives, priorities, and experiences of the participants, and some roles may have been underrepresented. In addition, recall bias and differences in terminology between stakeholder groups can influence the reported challenges and needs. Although interviews are appropriate for exploring workflows and requirements, the results would potentially be strengthened by complementary methods such as observations of real work practices or a longer-term pilot evaluation.

The literature review introduces further potential limitations. Literature searches always risk being too narrow, which may lead to relevant frameworks or alternative approaches being overlooked. In this thesis, a substantial part of the references comes from U.S. military standards, handbooks, and guidance documents. These sources were used because they are foundational for many of the core concepts in RAM and are frequently referenced in textbooks and academic literature in the field. However, it is important to recognise that such guidance often reflects a procurement perspective. In other words, it is typically written from the buyer's side of a contract, while Saab Naval operates closer to the supplier side. This perspective difference is not necessarily problematic in this context, since both parties generally share the goal of improving reliability and long-term sustainment outcomes. Still, the underlying viewpoint is worth keeping in mind when interpreting the literature-based reasoning and recommendations.

Finally, the thesis is limited by its theoretical framing and the selection of specific models and standards. The work primarily builds on RAM and

FRACAS-related concepts and the S5000F specification, as these were assessed as most relevant for the studied context and stakeholder needs. At the same time, there are additional frameworks, standards, and modelling approaches that could have been emphasised differently, which may influence the interpretation of what is most appropriate. Although the thesis aimed to consider alternative approaches where feasible, some degree of selection bias is difficult to avoid in a study with limited time and scope.

7.3 Research Contribution

This thesis contributes to the digitalisation of RAM-related feedback by combining an empirical study in a defence-industrial setting with the development of a web application. The work links established RAM and FRACAS principles with practical constraints observed at Saab Naval, and translates these insights into a web-based tool intended to support structured reporting, traceability, and analyses of operational data. The main contributions can be summarised as follows.

First, this thesis contributes empirical insight into how RAM-relevant feedback is produced, transformed, and lost across organisational settings in a secure, long-life engineering environment. Beyond describing a specific situation, the results highlight a set of recurring practical challenges that are likely to appear in similar settings: differences in incentives between data producers and data consumers (input and output users), unclear responsibilities, and challenges in turning operational observations into structured information that is reusable for analysis. This improves the understanding of why RAM feedback loops often degrade in practice, even if established processes and standards exist.

Second, this thesis contributes a structured translation of RAM/FRACAS principles into requirements suitable for digital reporting and analysis support. The value is not the specific list of fields, but the reasoning about what needs to be structured, what can remain flexible, and what trade-offs are required to balance analytical usefulness with reporting effort, from different stakeholder perspectives. This provides a design-oriented contribution that can

inform similar tool development efforts, especially where RAM analyses and traceability are desired but operational reporting time is limited.

Third, the functional web application serves as a concrete instantiation of these requirements and enables testable hypotheses about digital RAM feedback. By operationalising the proposed workflow, the thesis demonstrates feasibility and makes the discussion of data structures, roles, and traceability actionable rather than purely conceptual. The developed tool therefore functions as a research vehicle. It allows future work to evaluate data quality, adoption, and decision impact under realistic conditions, and it can be used as a baseline for iterative refinement and comparative evaluation.

Finally, the thesis contributes design-relevant insights that are particularly relevant for secure and confidentiality-driven areas. It clarifies that governance, role definitions, and technical integration are not secondary “deployment tasks”, but core design constraints that shape what a digital RAM tool can achieve and how it must be evaluated. By explicitly framing these constraints, the work supports more realistic research and development efforts, and it strengthens the bridge between standard-driven recommendations and industrial adoption.

Together, these contributions provide knowledge and a usable foundation for continued research on digital RAM feedback loops, including operational validation, refinement of metrics for adoption, and extension towards broader S-Series suite-aligned applications.

7.4 Future Work

Future work should focus on implementing and validating the digital RAM tool in real operations at Saab Naval. This thesis has delivered a functional web application whose design is informed by both stakeholder input and relevant theory. However, the solution has not yet been deployed in operations, hence several practical questions therefore remain. These include usability in daily work, data quality over time, and fit with existing processes and systems. Further work is needed to move from a prototype to sustained use and to assess measurable benefits for RAM analysis and decision-making.

7.4.1 Pilot Implementation and Evaluation

A suitable next step would be a pilot deployment in an operational setting, aiming to evaluate the tool during real use. The pilot should include users from both data input and data output roles and it needs to run long enough to capture normal variation in work and maintenance events. The pilot evaluation can focus on three main areas: reporting effort and user acceptance, data completeness and consistency, and analytical usefulness. Reporting effort can be assessed by measuring the time required to register an event and by collecting user feedback on whether the tool feels helpful or disruptive in daily work. Data completeness and consistency can be evaluated by reviewing how often key fields are populated, how reliably categories are applied, and where users still rely on free text. Analytical usefulness can then be assessed by examining whether the collected dataset supports meaningful analysis, such as identifying recurring failure patterns, key drivers of downtime, and maintainability-related issues. In addition, the pilot should assess traceability across the workflow. This includes whether operational observations can be linked to analysis outcomes and to follow-up actions, such as change requests or updates to maintenance procedures. The results from the pilot can then be used to refine the solution in an iterative way. This may involve improving the user interface, adjusting data definitions, and establishing governance routines for ownership and continuous feedback. Finally, the pilot findings can provide a basis for deciding if, and how, the solution should be scaled to additional teams, products, or systems at Saab Naval.

To enable the type of operational pilot described above, the web application must first be instantiated into Saab Naval's secure IT environment. The data required to use the tool in practice is confidential and cannot be handled outside approved systems and networks. A prerequisite for any real-world evaluation is therefore to establish a deployment setup that complies with internal security policies, including hosting, authentication, access control, and secure data storage. This integration work needs to be carried out by appropriate IT and information security personnel. Without it, the solution can only be evaluated

as a stand-alone prototype, which limits realism and prevents testing with actual operational data.

In parallel with the pilot, it would be beneficial to define and visualise a small set of KPIs in simple dashboards. The KPIs should focus on adoption, data usability, and realised impact. This includes reporting volume over time, reporting quality over time (i.e. how complete and well-structured problem reports are), and the number of concrete actions or changes initiated as a result of analyses performed in the tool. Together, these measures provide a pragmatic basis for evaluating whether the tool is used consistently, produces data that can support analysis, and leads to tangible improvements in practice.

To make these KPIs credible and sustainable, roles and governance around the tool must be defined clearly. This includes deciding who should act as system administrator(s), who is allowed to submit and edit reports, and who should have access as output users for analysis and decision support. Clear responsibilities are also needed for maintaining key data definitions and classifications, as well as handling updates when structures or categories need to be changed. By establishing access control and ownership early, Saab Naval can protect sensitive information, keep the dataset consistent, and ensure that the tool supports decision-making rather than becoming an uncontrolled reporting channel.

7.4.2 Scaling and Further Development

A further area for future work is to strengthen S-Series suite compliance and evolve the tool accordingly. The current implementation is primarily aligned with S5000F, based on what was considered most relevant for the studied workflow. Over time, the tool could be reworked to support the broader S-Series suite framework to a greater extent. This would increase standard alignment and improve interoperability with related processes, for example by extending data structures and traceability to better match the expectations across the S-Series specifications.

Another important direction is to expand the scope beyond AUV62-AT and

AUV62-MR. The tool is designed to be configurable and can be adapted to other products with similar reporting needs. A realistic, gradual rollout approach could be to first extend it to additional products within Underwater Systems, then to other business units within Saab Naval, and then potentially to other business areas within Saab. Such scaling would increase the value of the tool by enabling a more consistent feedback and analysis approach across products and organisational units, but it would also require careful configuration management and governance.

Future work could also include further expansion of the depth of the RAM perspective covered in this thesis. While the proposed solution addresses reliability, availability, and maintainability, the implemented analyses place most emphasis on reliability, through both basic metrics (e.g. MTBF/MDBF) and more advanced trend and distribution modelling using Duane and Weibull methods. A natural next step could therefore be to strengthen the maintainability dimension by incorporating additional maintenance-related data and metrics, such as calculating MDT and evaluating the effect of maintenance intervals. This would enable more concrete decision support for maintenance planning, including recommendations of suitable scheduled maintenance intervals and the identification of components that should be treated as potential spare parts based on observed failure behaviour and operational impact.

As the tool is scaled and used by a broader set of stakeholders, there could also be value in integrating it with established data visualisation platforms, such as Power BI. This would enable more polished dashboards and tailored reporting views, which can be useful when communicating findings to customers during meetings or when presenting analysis results as decision support to internal management. While advanced presentation capabilities are not a core requirement of the tool, the current database choice and structured data model provide a good foundation for such integrations.

Lastly, the tool can be further developed over time as new needs emerge and as adoption increases within the organisation. The current functionality reflects what the interviewed stakeholders and the literature study identified as most

relevant for an initial implementation. However, as more users begin to rely on the tool and the dataset grows, additional features may become valuable. One example is broadening the perspective beyond RAM to include related frameworks such as RAM-C/RAMCT or RAMS, by incorporating cost and safety aspects, and potentially other dimensions such as capability, testability, or similar system attributes. Such extensions should be introduced incrementally, based on demonstrated demand and the organisation's maturity in using the tool for operational feedback and decision support.

8 Conclusion

This thesis has addressed Saab Naval's need for a structured and implementable way to collect, structure, and use RAM-related in-service data for AUVs. The work was motivated by the current absence of a coherent digital RAM tool, despite a clear need for long-term traceability, consistent follow-up, and decision support related to RAM and its connection to LCC. Based on this background, this thesis set out to investigate how established RAM methodology and relevant standards, in particular S5000F, can be translated into a practical digital tool suitable for a secure industrial IT environment.

The results show that such a translation is feasible, but that it requires deliberate trade-offs between methodological completeness, usability, and implementability. While S5000F provides a valuable foundation for structuring in-service data feedback and defining relevant information categories, it cannot be applied without adaptation. The AUV context, the intended users, and the constraints of Saab Naval's secure IT environment all necessitate a more focused and operational interpretation. Consequently, the developed tool should be understood not as a full implementation of the standard, but as an adapted instantiation of selected principles judged most relevant and realistic within the scope of the thesis.

A central outcome of the work is therefore not only the tool itself, but also the accompanying structure for how RAM data can be captured, validated, related to product structures, and presented in a way that supports future analysis. In this sense, the thesis contributes with both a practical artefact and a structured framework for digital RAM application in a security-sensitive AUV setting. The work thereby provides Saab Naval with a concrete basis for continued development, while also illustrating how established RAM and FRACAS thinking can be operationalised in a context where commercial tools are not directly suitable.

At the same time, the thesis also demonstrates that the value of such a tool depends on more than software alone. Long-term usefulness requires organisational ownership, consistent reporting practices, clear data governance, and continued refinement of both input quality and analytical functionality. Since the thesis was limited to concept development and prototyping, questions related to deployment, integration into Saab Naval's IT infrastructure, long-term maintenance, and full-scale organisational adoption remain open. These aspects are critical if the proposed solution is to move from a promising prototype to an effective operational capability.

In conclusion, the thesis indicates that the potential value of a digital RAM tool for Saab Naval's AUVs is considerable, particularly in terms of improved structure, traceability, and decision support for RAM-related work. However, the work also shows that achieving this value in practice depends on successful continuation beyond the prototype stage, where technical implementation must be combined with structured organisational adoption and ongoing data quality control. The developed digital tool should therefore be seen as a foundation for future instantiation and evolution rather than as a finished end-state. In that respect, the thesis has taken an important step toward enabling more systematic and long-term RAM work for AUVs within Saab Naval.

BIBLIOGRAPHY

- [1] S. Saraswat and G. Yadava, “An overview on reliability, availability, maintainability and supportability (rams) engineering,” *International Journal Quality & Reliability Management*, vol. 25, no. 3, pp. 330–344, Mar. 2008, ISSN: 0265-671X. DOI: 10.1108/02656710810854313. eprint: <https://www.emerald.com/ijqrm/article-pdf/25/3/330/995231/02656710810854313.pdf>. [Online]. Available: <https://doi.org/10.1108/02656710810854313>.
- [2] SEBoK Editorial Board. “System reliability, availability, and maintainability.” SEBoK v. 2.13 (released 2025-11-17). Lead authors: Paul Phister and David Olwell, Accessed: Feb. 3, 2026. [Online]. Available: https://sebokwiki.org/wiki/System_Reliability,_Availability,_and_Maintainability.
- [3] J. Zhang, C. Haskins, Y. Liu, and M. A. Lundteigen, “A systems engineering-based approach for framing reliability, availability, and maintainability: A case study for subsea design,” *Systems Engineering*, vol. 21, no. 6, pp. 576–592, 2018. DOI: <https://doi.org/10.1002/sys.21462>. eprint: <https://incose.onlinelibrary.wiley.com/doi/pdf/10.1002/sys.21462>. [Online]. Available: <https://incose.onlinelibrary.wiley.com/doi/abs/10.1002/sys.21462>.
- [4] Secretary of the Navy, “Reliability and maintainability engineering guidebook,” Department of the Navy, Tech. Rep., Jun. 2022. Accessed: Feb. 16, 2026. [Online]. Available: <https://www.cto.mil/wp-content/uploads/2023/09/SECNAV-RME-2022.pdf>.
- [5] U.S. Government Accountability Office, “Navy shipbuilding: Increasing focus on sustainment early in the acquisition process could save billions,” U.S. Government Accountability Office, Washington, DC, Tech. Rep. GAO-20-2, Mar. 2020, Published: 2020-03-24. Accessed: 2026-02-03.

- [Online]. Available: <https://www.gao.gov/assets/gao-20-2.pdf>.
- [6] “MIL-STD-2155: Failure reporting, analysis and corrective action system (FRACAS),” U.S. Department of Defense, Military Standard MIL-STD-2155, Jul. 24, 1985, Superseded by MIL-HDBK-2155 (1995). Accessed: Feb. 18, 2026. [Online]. Available: <https://d6scj24zvfbbo.cloudfront.net/bd896af554575015c05ebd69aae524ae/200000352-0ca680ca6b/MIL-STD-2155.pdf>.
- [7] “MIL-HDBK-189C: Reliability growth management,” Department of Defense, Handbook MIL-HDBK-189C, Jun. 14, 2011. Accessed: Feb. 18, 2026. [Online]. Available: <https://www.dote.osd.mil/Portals/97/docs/TEMPGuide/MIL-HDBK-189C.pdf>.
- [8] Aerospace Industries Association (AIA), “Aia s-series support for mbe,” Aerospace Industries Association (AIA), Tech. Rep., Sep. 2023, White paper. Published: 2023-09-07. Accessed: 2026-02-03. [Online]. Available: <https://www.aia-aerospace.org/wp-content/uploads/AIA-S-Series-Support-for-MBE-White-Paper-1.pdf>.
- [9] *S5000f: International specification for in-service data feedback*, Copyright © Aerospace, Security and Defence Industries Association of Europe (ASD). Part of the IPS S-Series™ suite of specifications, Integrated Product Support (IPS) S-Series™, 2025.
- [10] Saab AB. “Organisation | company in brief.” Web page, Accessed: Feb. 3, 2026. [Online]. Available: <https://www.saab.com/about/company-in-brief/organisation>.
- [11] Wikipedia contributors. “Saab kockums.” Wikipedia, the free encyclopedia, Accessed: Feb. 3, 2026. [Online]. Available: https://en.wikipedia.org/wiki/Saab_Kockums.

- [12] Saab AB. “Saab completes acquisition of tkms ab (kockums).” Press release, Accessed: Feb. 3, 2026. [Online]. Available: <https://www.saab.com/newsroom/press-releases/2014/saab-completes-acquisition-of-tkms-ab-kockums>.
- [13] Saab AB. “Naval | products.” Web page, Accessed: Feb. 3, 2026. [Online]. Available: <https://www.saab.com/products/naval>.
- [14] Saab AB, “Auv62-at: System overview,” Saab AB, Tech. Rep., Feb. 2020, Brochure, AUV62-AT – EN – ver. 31, February 2020. Accessed: 2026-02-03. [Online]. Available: <https://www.saab.com/contentassets/f55ba5f6d2824433b76b65e5745244fd/saab-auv62-at-brochure.pdf>.
- [15] Saab AB, “Auv62-mr: Mine reconnaissance system,” Saab AB, Tech. Rep., Sep. 2015, Brochure, AUV62-MR, English, ver. 1, September 2015. Accessed: 2026-02-03. [Online]. Available: <https://www.saab.com/globalassets/event/aeroindia/auv62-mr-mine-reconnaissance-system.pdf>.
- [16] Saab AB. “Saab brand & media portal (image asset).” Image accessed via Saab Brand Portal; usage permitted according to portal guidelines., Accessed: Mar. 25, 2026. [Online]. Available: https://brand.saab.com/point/en/saab/component/default/25854#1422262171509_0.
- [17] M. Höst, B. Regnell, and P. Runeson, *Att genomföra examensarbete*. Studentlitteratur, 2006.
- [18] M. Denscombe, *Forskningshandboken, För småskaliga forskningsprojekt inom samhällsvetenskaperna*, Swedish, 4th ed., trans. by P. Larson. Lund: Studentlitteratur, 2018, ISBN: 978-91-44-12288-5.
- [19] G. Kovács and K. M. Spens, “Abductive reasoning in logistics research,” *International Journal of Physical Distribution & Logistics Management*, vol. 35, no. 2, pp. 132–144, 2005. DOI: 10.1108/096000030510590318.

- [20] A. Bryman, *Brymans samhällsvetenskapliga forskningsmetoder*, 4th ed. Stockholm: Liber, 2025, Svensk översättning av *Bryman's Social Research Methods* (6th ed., 2021, Oxford University Press), ISBN: 978-91-47-14701-4.
- [21] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, Mar. 2004. DOI: 10.2307/25148625.
- [22] U.S. Department of Defense, "DoD Instruction 5000.91: Product support management for the adaptive acquisition framework," Office of the Under Secretary of Defense for Acquisition and Sustainment, DoD Instruction DoDI 5000.91, Dec. 8, 2021. Accessed: Feb. 23, 2026. [Online]. Available: <https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500091p.PDF>.
- [23] Defense Acquisition University. "Life cycle cost," Defense Acquisition University, Accessed: Feb. 23, 2026. [Online]. Available: <https://www.dau.edu/glossary/life-cycle-cost>.
- [24] Office of the Secretary of Defense, Cost Assessment and Program Evaluation (CAPE), "Operating and support cost-estimating guide," U.S. Department of Defense, Guidebook, 2025. Accessed: Feb. 23, 2026. [Online]. Available: <https://www.dau.edu/sites/default/files/2025-02/2025%20OS%20Cost%20Estimating%20Guide.pdf>.
- [25] U.S. Government Accountability Office, "Gao cost estimating and assessment guide: Best practices for developing and managing program costs," U.S. Government Accountability Office, Guide GAO-20-195G, Mar. 1, 2020. Accessed: Feb. 23, 2026. [Online]. Available: <https://www.gao.gov/assets/gao-20-195g.pdf>.
- [26] J. V. Jones, *Integrated Logistics Support Handbook*, 3rd. New York, NY, USA: McGraw-Hill Education, 2006.

- [27] Department of Defense, “Acquisition of support equipment and associated integrated logistics support,” U.S. Department of Defense, Tech. Rep. MIL-HDBK-2097A, 1997. Accessed: Feb. 26, 2026. [Online]. Available: <https://quicksearch.dla.mil/Transient/A004F127D5E1458E9092A22F182DACBE.pdf>.
- [28] *Space systems – integrated logistic support*, International Organization for Standardization, Aug. 2018. Accessed: Feb. 26, 2026. [Online]. Available: <https://cdn.standards.iteh.ai/samples/72937/0d526ff45d6e4111bfe18d4bc3b5569c/ISO-16091-2018.pdf>.
- [29] G. Desiderio, “Acquisition and management of integrated logistic support for systems and equipment,” Assistant Secretary of Defense (Production and Logistics), Washington, DC, USA, Tech. Rep. DoD-D-5000.39, Nov. 1983.
- [30] S. J. C. s. Gordon G. S. Merchant, “Integrated logistics support guide,” Defense Systems Management College, Fort Belvoir, VA, USA, Tech. Rep. AD-A282 913, Jun. 1994. Accessed: Feb. 26, 2026. [Online]. Available: <https://apps.dtic.mil/sti/tr/pdf/ADA282913.pdf>.
- [31] M. Willis. “What went wrong with ils and can ips fix it?” Accessed: Feb. 26, 2026. [Online]. Available: https://www.pennantplc.com/ils_ips_article/.
- [32] Aerospace, Security and Defence Industries Association of Europe (ASD). “About sx000i,” Accessed: Feb. 26, 2026. [Online]. Available: <https://www.s-series.org/sx000i/about-sx000i/>.
- [33] Defense Acquisition University, “Integrated product support (ips) elements guidebook,” Defense Acquisition University, Tech. Rep., Jan. 2024.
- [34] D. J. Smith, *Reliability, Maintainability and Risk: Practical Methods for Engineers*, 7th ed. Butterworth-Heinemann, 2005.

- [35] R. F. Stapelberg, *Handbook of Reliability, Availability, Maintainability and Safety in Engineering Design*. London: Springer, 2009, ISBN: 9781848001749. DOI: 10.1007/978-1-84800-175-6.
- [36] “MIL-STD-721C: Definitions of terms for reliability and maintainability,” U.S. Department of Defense, Military Standard MIL-STD-721C, Jun. 10, 1981. Accessed: Feb. 16, 2026. [Online]. Available: https://fa.reliability-risk.com/wp-content/uploads/2024/02/mil_std_721c.pdf.
- [37] M. Reid. “How are the plotting positions calculated.” Version 0.9.0 documentation, reliability documentation, Accessed: Apr. 18, 2026. [Online]. Available: <https://reliability.readthedocs.io/en/stable/How%20are%20the%20plotting%20positions%20calculated.html>.
- [38] P. D. T. O’Connor and A. Kleyner, *Practical Reliability Engineering*, 5th ed. Chichester, United Kingdom: John Wiley & Sons, 2012, ISBN: 978-0-470-97982-2.
- [39] Z. Yang, J. Kannianen, T. Krogerus, and F. Emmert-Streib, “Prognostic modeling of predictive maintenance with survival analysis for mobile work equipment,” *Scientific Reports*, vol. 12, p. 8529, 2022. DOI: 10.1038/s41598-022-12572-z. Accessed: Mar. 17, 2026. [Online]. Available: <https://www.nature.com/articles/s41598-022-12572-z>.
- [40] Z. Kharazian, T. Lindgren, S. Magnússon, O. Steinert, O. Andersson Reyna, et al., “Scania component x dataset: A real-world multivariate time series dataset for predictive maintenance,” *Scientific Data*, vol. 12, p. 493, 2025. DOI: 10.1038/s41597-025-04802-6. Accessed: Mar. 17, 2026. [Online]. Available: <https://www.nature.com/articles/s41597-025-04802-6>.
- [41] B. Coutinho, M. Moreira, E. Pereira, and G. Gonçalves, “Survival analysis-based system for predictive maintenance optimization,” *SN Computer Science*, vol. 6, p. 766, 2025. DOI: 10.1007/s42979-

- 025-04291-9. Accessed: Mar. 17, 2026. [Online]. Available: <https://link.springer.com/article/10.1007/s42979-025-04291-9>.
- [42] M. Rahat, Z. Kharazian, P. Sheikholharam Mashhadi, T. Rönngvaldsson, and S. Choudhury, “Bridging the gap: A comparative analysis of regressive remaining useful life prediction and survival analysis methods for predictive maintenance,” in *Proceedings of the Asia Pacific Conference of the PHM Society 2023*, vol. 4, 2023. doi: 10.36001/phmap.2023.v4i1.3646. Accessed: Mar. 17, 2026. [Online]. Available: <https://papers.phmsociety.org/index.php/phmap/article/view/3646>.
 - [43] M. Zhang, C. Cui, S. Liu, and X. Yi, “Reliability technology using FTA, FMECA, FHA and FRACAS: A review,” in *2021 IEEE International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)*, IEEE, 2021, pp. 282–291. doi: 10.1109/SDPC52933.2021.9563512.
 - [44] Aerospace, Security and Defence Industries Association of Europe (ASD). “The S-Series™ Integrated Product Support (IPS) Specifications,” Accessed: Feb. 17, 2026. [Online]. Available: <https://www.s-series.org>.
 - [45] *S3000l: International procedure specification for logistics support analysis*, Copyright © Aerospace, Security and Defence Industries Association of Europe (ASD). Part of the IPS S-Series™ suite of specifications, Integrated Product Support (IPS) S-Series™, 2025.
 - [46] Aerospace, Security and Defence Industries Association of Europe (ASD). “Resource center,” Accessed: Feb. 24, 2026. [Online]. Available: <https://www.s-series.org/resource-center/>.

APPENDICES

Appendix A: Interview Guide

Appendix A presents the interview guide used for the semi-structured interviews conducted with RAM/ILS experts and prospective output and input users.

1. Introduction and Context (approx. 5 min)

- Could you briefly describe your role and how you currently work with RAM/ILS?
- In which situations do you need, or could you benefit from, RAM-related information or analyses?

2. Introduction to the Tool/Thesis Setup (approx. 5-10 min)

- We present the basic concept of the tool and the core functionalities already identified.
- Opportunity for initial questions and clarifications.

3. Desired Outputs and Use Cases (approx. 15-20 min)

- How would you envision such a tool in terms of design and functionality?
- What are the most important results the tool should provide for you as an output/input user?
- Which RAM metrics/KPIs are most relevant for your work (and why)?
- What types of reports or visualisations would you need or prefer?

4. Data Requirements and Structure (approx. 10-15 min)

- The export function: what should its purpose and scope be?
- What data should be exported? All data, or only selected values?
- Opportunity for more technical questions related to input data.

5. Requirements and Prioritisation (approx. 5-10 min)

- If the first version of the tool could only include three functions, which would they be?
- What is most important for you to actually use the tool (acceptance criteria)?
- Do you see any risks associated with introducing a new tool (e.g. misinterpretation, low data quality, increased workload)?

6. Closing (approx. 2-3 min)

- Are there other people or roles you think we should speak to in order to capture additional perspectives?
- Is there anything important that you feel we have missed?

Appendix B: Notations

Appendix B summarises the notations used throughout the thesis, including definitions of key symbols and parameters.

- $\hat{\lambda}$: failure rate (hazard rate)
- k : number of observed failures
- T : total operating time
- N_r : number of corrective actions
- $t_{\text{repair},i}$: active repair time for the i th corrective maintenance action
- N_o : number of outages
- $t_{\text{down},i}$: total downtime for the i th failure event
- $\bar{A}$: unavailability or PFD
- β : Weibull shape parameter
- η : Weibull scale parameter (characteristic life)
- $f(t)$: probability density function at time t
- $R(t)$: reliability function at time t
- $h(t)$: failure rate (hazard) function at time t
- t : cumulative test time
- K : Duane-model constant (scaling parameter)
- α : reliability growth rate

Appendix C: Interviewees

Appendix C provides an overview of the interviewees included in the study summarising their roles at Saab Naval and the type of interview conducted, see Table C.1 below.

Table C.1: List of interviewees' role at Saab Naval and the type of interview.

Number	Role	Type of interview
I1	ILS-Engineer	Semi-structured & unstructured interviews (hybrid)
I2	Systems Engineer	Semi-structured & unstructured interviews (hybrid)
I3	ILS-Engineer	Semi-structured interview (online)
I4	ILS-Engineer	Semi-structured interview (online)
I5	Cloud Engineer	Unstructured interviews (online)
I6	Software Developer	Unstructured interviews (online)

Appendix D: Product Requirements Document

Appendix D contains the PRD, which summarises the functional and non-functional requirements derived for the digital RAM tool.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

PRD – Product Requirements Document

Oscar Cedell | Master Thesis | oscar.cedell@saabgroup.com

Jacob Persson | Master Thesis | jacob.persson@saabgroup.com

Revision History

Date	Version	Description	Author
2026-03-04	0.1	Initial disposition	Oscar Cedell
2026-03-04	0.2	Introduction chapter	Oscar Cedell
2026-03-04	0.3	Background chapter	Oscar Cedell
2026-03-10	0.4	EP-01 User Registration	Oscar Cedell
2026-03-10	0.5	EP-02 System Configuration	Oscar Cedell
2026-03-10	0.6	EP-03 User Management	Oscar Cedell
2026-03-10	0.7	EP-04 Product Context Selection	Oscar Cedell
2026-03-11	0.8	Initial Excel File Management chapter	Oscar Cedell
2026-03-11	0.9	EP-05 Problem Report Import	Oscar Cedell
2026-03-11	0.10	EP-06 Problem Report Management	Oscar Cedell
2026-03-13	0.11	EP-08 Action Report Management	Oscar Cedell
2026-03-13	0.12	Visitor Interface Concepts	Oscar Cedell
2026-03-13	0.13	Authorisation Interface Concepts	Oscar Cedell
2026-03-13	0.14	Landing Page Interface Concepts	Oscar Cedell
2026-03-13	0.15	Admin Interface Concepts	Oscar Cedell
2026-03-15	0.16	Writer Interface Concepts	Oscar Cedell
2026-03-15	0.17	Problem Report Management Interface Concepts	Oscar Cedell

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

2026-03-15	0.18	Action Report Management Interface Concepts	Oscar Cedell
2026-03-18	0.19	7 Excel File Management chapter	Jacob Persson
2026-03-18	0.20	EP-07 RAM Analysis	Oscar Cedell
2026-03-18	0.21	Analysis Interface Concepts	Oscar Cedell
2026-03-19	0.22	EP-09 Export Data	Oscar Cedell
2026-03-19	0.23	EP-10 Authorisation Warnings	Oscar Cedell
2026-03-19	0.24	8 Action Report Data Definition	Oscar Cedell
2026-03-20	0.25	5 Non-Functional Requirements	Oscar Cedell
2026-03-20	0.26	6 User Interface descriptions	Oscar Cedell
2026-03-20	0.27	10 RAM Metric Definitions... initial chapter description	Oscar Cedell
2026-03-20	1.0	Review	Jacob Persson
2026-03-30	1.1	“Viewer” role to “Analyst”	Oscar Cedell
2026-03-30	1.2	Action Report creation requires Editor/Admin role	Oscar Cedell
2026-03-30	1.3	Added priority description	Jacob Persson
2026-03-30	1.4	Epic-02 rewrite	Jacob Persson
2026-03-31	1.5	US-05-04, prevent import of duplicate report	Jacob Persson
2026-03-31	1.6	US-03-04, require password reset, after admin reset	Jacob Persson
2026-04-06	1.7	Replaced module wireframes with breakdown items	Oscar Cedell

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

2026-04-13	1.8	Updated Introduction chapter	Jacob Persson
2026-04-13	1.9	Added User role to background chapter	Jacob Persson
2026-04-13	1.10	Added US-07-08 Units overview	Oscar Cedell
2026-04-13	1.11	Excluded (former) chapter 3 System Use Case Design	Oscar Cedell
2026-04-14	1.12	Added figure text on all interface concepts	Jacob Persson
2026-04-14	1.13	Small changes in 5.6.2 and 5.6.11	Jacob Persson
2026-04-14	1.14	Reworked chapter 9 Analysis Calculations	Jacob Persson
2026-04-14	1.15	Completed chapter 8 S5000F Export Data Design	Oscar Cedell
2026-04-15	2.0	Review	Oscar Cedell

Content

1 Introduction.....	135
2 Background.....	135
2.1 Purpose of the System.....	136
2.2 System Users.....	136
3 Epics, User Stories, and Functional Requirements.....	137
EP-01 User Registration and Authentication.....	138
US-01-01 Access authentication interface.....	138
US-01-02 Register as a new user.....	138
US-01-03 Sign in as a user.....	139
US-01-04 Sign out as a user.....	140
US-01-05 Initial admin registration.....	141
EP-02 System Configuration.....	141
US-02-01 Manage products (Admin).....	141
US-02-02 Manage product units (Admin).....	142
US-02-03 Manage product breakdown items (Admin).....	142
US-02-04 Manage predictions (Admin).....	143
US-02-05 Manage targets (Admin).....	144
US-02-06 Manage failure areas (Admin).....	145
EP-03 User Management.....	145
US-03-01 View list of users (Admin).....	145
US-03-02 Search, filter, and sort users (Admin).....	146
US-03-03 Manage user account information (Admin).....	147
US-03-04 Assign user role (Admin).....	147
US-03-05 Assign user products (Admin).....	148
US-03-06 Reset user password (Admin).....	148
US-03-07 Disable user account (Admin).....	149
EP-04 Product Context Selection.....	149
US-04-01 Select active product.....	149
US-04-02 Persist selected product.....	150
EP-05 Problem Report Import.....	150

US-05-01 Upload problem report dataset.....	151
US-05-02 Map and validate uploaded data.....	151
US-05-03 Confirm import.....	152
EP-06 Problem Report Management.....	153
US-06-01 View list of problem reports.....	153
US-06-02 Search, filter, and sort problem reports.....	154
US-06-03 View problem report details.....	155
US-06-04 Disable a problem report.....	155
EP-07 RAM Analysis.....	156
US-07-01 Filters for analysis.....	156
US-07-02 Visualise failure data distribution.....	157
US-07-03 Visualise failure occurrence over time.....	158
US-07-04 Display calculated RAM metrics.....	158
US-07-05 Display predicted and targeted RAM metrics.....	159
US-07-06 Visualise cumulative average trend.....	160
US-07-07 Visualise reliability trends.....	161
US-07-08 Units overview.....	162
EP-08 Action Report Management.....	163
US-08-01 Create a new action report.....	163
US-08-02 View list of action reports.....	164
US-08-03 Search, filter, and sort action reports.....	165
US-08-04 View action report details.....	165
US-08-05 Disable an action report.....	166
US-08-06 Edit an action report.....	166
US-08-07 Complete an action report.....	167
EP-09 Export Data.....	167
US-09-01 Export RAM metrics.....	167
US-09-02 Export action report.....	168
EP-10 Authorisation Warnings.....	169
US-10-01 Display warning for user without assigned role and/or product....	169
US-10-02 Display warning for user without a selected product.....	170
US-10-03 Display warning for unauthorised access.....	170

4 Non-Functional Requirements.....	171
4.1 Security.....	171
4.2 Compliance.....	172
4.3 Performance.....	172
4.4 Usability.....	173
4.5 Maintainability.....	173
4.6 Availability.....	173
5 User Interface Concepts.....	174
5.1 Visitor Interfaces.....	174
5.1.1 Sign In.....	174
5.1.2 Register.....	175
5.2 General User Interfaces.....	176
5.2.1 Landing Page.....	176
5.3 Authorisation Warnings.....	176
5.3.1 Missing Role/Product Warning.....	176
5.3.2 Missing Product Selection Warning.....	177
5.3.3 Unauthorised Access Warning.....	178
5.4 Admin Interfaces.....	178
5.4.1 Top Navigation.....	178
5.4.2 Product Management.....	179
5.4.3 Breakdown Item Management.....	179
5.4.4 Unit Management.....	180
5.4.5 Product Predictions Management.....	181
5.4.6 Target Management.....	182
5.4.7 Failure Area Management.....	182
5.4.8 User List.....	183
5.4.9 User Management.....	183
5.4.10 Reset Password.....	185
5.5 Reporter Interfaces.....	185
5.5.1 Top Navigation.....	185
5.5.2 Start Import Problem Report Flow.....	186
5.5.3 Display Problem Report Validation Errors.....	186
5.5.4 Inspect Uploaded Problem Report.....	188

5.5.5 Display Problem Report Import Error.....	190
5.5.6 Display Problem Report Import Confirmation.....	190
5.6 Analyst Interfaces.....	191
5.6.1 Top Navigation.....	191
5.6.2 Problem Report List.....	192
5.6.3 Detailed Problem Report.....	192
5.6.4 Analysis Filter Component.....	194
5.6.5 Failure Data Distribution Visualisation.....	194
5.6.6 Failure Occurrence Trend.....	195
5.6.7 RAM Metrics Analysis.....	196
5.6.8 Reliability Trends.....	197
5.6.9 Action Report List.....	198
5.6.10 New Action Report Form.....	198
5.6.11 Detailed Action Report.....	199
6 Excel File Management.....	200
6.1 Excel File Structure.....	201
6.1.1 Problem Report Form.....	201
6.1.2 Revision History.....	203
6.1.3 Reference Data.....	204
6.2 Excel File Archivation.....	204
7 Action Report Data Definition.....	205
8 S5000F Data Export Design.....	206
8.1 RAM Metrics Export.....	207
8.2 Action Report Export.....	208
9 Analysis Calculations.....	209
9.1 Aggregation of Filters.....	210
9.2 RAM Metrics.....	210
9.2.1 MTBF, MDBF, MTTR and Availability.....	210
9.3 Reliability Trends.....	212
9.3.1 Duane.....	212
9.2.2 Weibull.....	214

Terminology

This section defines all terminology, abbreviations and acronyms used in the document:

- Admin, Administrator – A user with elevated privileges, full system access
- Back-end – The server-side implementation, REST API, and database logic
- Discipline – A global engineering classification domain, such as Electrical/Electronic, Mechanical, or Software/Firmware
- Editor – A user which can read and write data
- EP – Epic
- Failure Area – A global sub-classification within a discipline, used to categorise reported issues or events
- FR – Functional Requirement
- Front-end – The web client and user interface components of the system
- LCN – Logistic Support Analysis Control Number
- MDBF – Mean Distance Before Failure
- MTBF – Mean Time Before Failure
- MTTR – Mean Time To Repair
- NFR – Non-Functional Requirement
- PRD – Product Requirements Document
- Product – A product type managed by the system
- RAM – Reliability Availability Maintainability
- RBAC – Role-Based Access Control
- Reporter – A user which can import problem reports
- US – User Story
- User – A user which does not have access to any data or system functionality.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

- Analyst – A user with read-only access
- XML – eXtensible Markup Language
- XSD – XML Schema Definition

1 Introduction

This document presents the product requirements for a digital RAM (Reliability, Availability, Maintainability) tool developed within the context of autonomous underwater vehicles (AUVs) for Saab Naval. The purpose of the system is to support structured collection, management, and analysis of RAM data for AUVs.

RAM analysis plays a critical role in decision-making related to system design, maintenance planning, and lifecycle cost optimisation. However, existing processes for handling RAM data are often fragmented or insufficiently supported. This creates challenges in data consistency, traceability, and analysis.

The proposed system aims to address these challenges by providing a unified platform for managing problem reports, action reports, and associated reliability data. Through this united management the system is able to support RAM analysis through trend visualisation and calculation of key metrics. Additionally, the system is designed with consideration for interoperability and standardisation, such as alignment with the S5000F specification for in-service data feedback.

This document defines the system requirements through epics, user stories, functional and non-functional requirements. They cover aspects such as user roles and their access control, system behaviour and functionality, and export capabilities.

2 Background

This chapter includes the purpose of the system and its intended users.

2.1 Purpose of the System

The purpose of the system is to provide a structured digital tool for managing and analysing RAM data for AUVs. The system supports the following core capabilities:

- Collection and storage of failure-related data through problem reports.
- Analysis of RAM through metrics, distributions and trend visualisations.
- Filtering and exploration of data to support analysis workflows.
- Documentation of decisions and corrective actions through action reports.
- Export of data in a standardised XML format aligned with S5000F.

2.2 System Users

The system implements role-based access control (RBAC), where permissions are assigned to roles and roles are assigned to users. There are five types of assignable roles: Administrator (Admin), Reporter, Analyst, Editor, and User.

- Admins are users with elevated privileges, such as assigning roles to users. Admins have complete system access, i.e. the access rights of all other roles on top of their own.
- Reporters are users with system access for input of failure reports.
- Analysts are users with system access for reading data and analyses.
- Editors are users with the combined system access of a Reporter and an Analyst, and creating action reports.
- Users are users which have registered to the application, but have not yet been assigned one of the prior mentioned roles. A User do not have access to any data or system functionality.

On top of these global roles, all users can be assigned specific product access which determines what product they have access to write or read data associated with.

3 Epics, User Stories, and Functional Requirements

This chapter describes the functional behaviour of the system through epics, user stories, and functional requirements.

Each epic represents a high-level functional area of the system and is identified by the prefix EP, followed by a sequential number (e.g. EP-01). Epics are presented as subsections of this chapter.

User stories (US) belonging to an epic are identified using the format US-XX-YY, where XX corresponds to the epic identifier and YY is the sequential identifier of the US within that epic. For example, US-01-02 refers to the second US belonging to EP-01.

Each US contains a set of functional requirements (FR) that define the system behaviour necessary to fulfil the user story. Functional requirements are identified using the format FR-XX-YY-ZZ, where XX refers to the epic, YY to the US, and ZZ is the sequential identifier of the functional requirement.

Each US is assigned a priority on a scale from 1 to 5, where 5 represents the highest priority. The assigned priority reflects the assessed relevance and urgency

of the functionality in relation to the purpose of the system and the scope of the project. Higher-priority US are those considered essential to the core functionality of the system and are typically supported by multiple inputs from stakeholder interviews, literature, and benchmarking. Lower-priority US represent functionality considered less critical in the initial scope, more specialised in nature, or more suitable for future extension.

EP-01 User Registration and Authentication

The system shall allow users to create accounts, authenticate themselves, and manage system access securely.

US-01-01 Access authentication interface

Priority: 5

Story:

As a visitor, I want to be presented with options to sign in or register so that I can access the system.

Functional Requirements:

FR-01-01-01 When accessing the system without an authenticated session, the system shall present options to sign in or register.

FR-01-01-02 The system shall ensure that authenticated users cannot access the authentication interface.

US-01-02 Register as a new user

Priority: 5

Story:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

As a new user, I want to be able to register an account using my email, name, and password so that I can access the system.

Functional Requirements:

FR-01-02-01 The system shall provide a registration form containing the fields:

FR-01-02-a Name

FR-01-02-b Email

FR-01-02-c Password

FR-01-02-02 The system shall validate that the provided email address is a unique @saabgroup.com adress.

FR-01-02-03 The system shall present and enforce the following password requirements:

FR-01-02-a Minimum length of 12 characters

FR-01-02-b At least one uppercase letter

FR-01-02-c At least one digit

FR-01-02-d At least one special character

FR-01-02-04 Upon successful registration, the system shall authenticate the user and grant access according to the User role.

FR-01-02-05 If registration fails, the system shall display an error message.

US-01-03 Sign in as a user

Priority: 5

Story:

As a registered user, I want to authenticate myself using my email and password so that I can access the system.

Functional Requirements:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-01-03-01 The system shall provide an authentication form containing the fields:

FR-01-03-a Email

FR-01-03-b Password

FR-01-03-02 The system shall display the process for requesting a new password.

FR-01-03-03 If the provided credentials are valid, the system shall authenticate the user and grant access to the system according to their assigned role.

FR-01-03-04 If the credentials are invalid, the system shall reject the authentication attempt and display an error message.

US-01-04 Sign out as a user

Priority: 5

Story:

As an authenticated user, I want to sign out of the system so that my account remains secure when I am no longer using the system.

Functional Requirements:

FR-01-04-01 The system shall provide a sign-out option accessible to authenticated users.

FR-01-04-02 The system shall terminate the current authenticated session when a user signs out.

FR-01-04-03 After signing out, the system shall redirect the user to the authentication interface.

FR-01-04-04 If a user attempts to access a restricted resource without a valid authenticated session, the system shall deny access and redirect the user to the authentication interface.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

US-01-05 Initial admin registration

Priority: 4

Story:

As a system administrator, I want the system to provide an initial Admin account so that the system can be configured after set up.

Functional Requirements:

FR-01-05-01 When the system is initialised, a default Admin account shall be created.

EP-02 System Configuration

This epic covers the administration of products and related configurations used when importing and analysing data.

US-02-01 Manage products (Admin)

Priority: 5

Story:

As an Admin, I want to create and modify products so that data can be associated with specific products for analysis.

Functional Requirements:

FR-02-01-01 The system shall allow Admins to create new products with the values:

FR-02-01-a Name

FR-02-01-02 The system shall allow Admins to modify existing products.

FR-02-01-03 The system shall allow Admins to mark a product as enabled or disabled.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-02-01-04 The system shall ensure each product has a unique name within the system.

US-02-02 Manage product units (Admin)

Priority: 5

Story:

As an Admin, I want to create and modify units for a specific product so that unit-level operational data can be defined and used for analysis.

Functional Requirements:

FR-02-02-01 The system shall allow Admins to create units associated with a specific product with the values:

FR-02-03-a Name

FR-02-02-02 The system shall ensure a unit is associated with exactly one product.

FR-02-02-03 The system shall allow Admins to view all created units for a product in a list.

FR-02-02-04 The system shall allow Admins to modify existing units.

FR-02-02-05 The system shall allow Admins to mark a unit as enabled or disabled.

FR-02-02-06 The system shall ensure each unit has a unique name within a specific product.

US-02-03 Manage product breakdown items (Admin)

Priority: 5

Story:

As an Admin, I want to create and modify breakdown items within a specific product so that data can be associated with the correct item in the product structure.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Functional Requirements:

FR-02-03-01 The system shall allow Admins to create breakdown items associated with a specific product with the values:

FR-02-03-a LCN

FR-02-03-b Name

FR-02-03-02 The system shall allow Admins to modify existing breakdown items.

FR-02-03-03 The system shall allow Admins to mark a breakdown item as enabled or disabled.

FR-02-03-04 The system shall ensure that a breakdown item is associated with exactly one product.

FR-02-03-05 The system shall ensure that a nested breakdown item references one parent breakdown item within the same product.

US-02-04 Manage predictions (Admin)

Priority: 5

Story:

As an Admin, I want to set and update predicted RAM metrics for a specific product, unit, or breakdown item so that they can be used as reference values for analysis.

Functional Requirements:

FR-02-04-01 The system shall allow Admins to set predicted RAM metrics for a specific product, unit, or breakdown item.

FR-02-04-02 The system shall allow Admins to input predicted values for:

FR-02-04-a MTBF

FR-02-04-b MDBF

FR-02-04-c MTTR

FR-02-04-d Availability

FR-02-04-03 The system shall allow Admins to modify existing predicted values.

FR-02-04-04 The system shall associate each set of predicted values with exactly one product, unit, or breakdown item.

FR-02-04-05 The system shall not require input of predictions on any level.

US-02-05 Manage targets (Admin)

Priority: 5

Story:

As an Admin, I want to set and update target RAM metrics for a specific product, unit, or breakdown item so that they can be used as reference values for analysis.

Functional Requirements:

FR-02-05-01 The system shall allow Admins to set target RAM metrics for a specific product, unit, or breakdown item.

FR-02-05-02 The system shall allow Admins to input target values for:

FR-02-05-a MTBF

FR-02-05-b MDBF

FR-02-05-c MTTR

FR-02-05-d Availability

FR-02-05-03 The system shall allow Admins to modify existing target values.

FR-02-05-04 The system shall associate each set of target values with exactly one product, unit, or breakdown item.

FR-02-05-05 The system shall not require input of targets on any level.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

US-02-06 Manage failure areas (Admin)

Priority: 5

Story:

As an Admin, I want to create and modify failure areas within a specific discipline so that data can be associated with specific failure areas for analysis.

Functional Requirements:

FR-02-06-01 The system shall allow Admins to create failure areas associated with one of the disciplines:

FR-02-06-a Electrical/Electronic

FR-02-06-b Mechanical

FR-02-06-c Software/Firmware

FR-02-06-02 The system shall allow Admins to modify existing failure areas.

FR-02-06-03 The system shall allow Admins to mark a failure area as enabled or disabled.

FR-02-06-04 The system shall ensure each failure area is associated with exactly one discipline.

FR-02-06-05 The system shall prevent creation of a failure area with the same name within a discipline.

EP-03 User Management

This epic covers the administration of user accounts, roles, and product access within the system.

US-03-01 View list of users (Admin)

Priority: 5

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Story:

As an Admin, I want to view a list of all registered users so that I can access and manage a specific user account.

Functional Requirements:

FR-03-01-01 The system shall allow Admins to view a list of all registered users.

FR-03-01-02 The user list shall display at least the following information for each user:

FR-03-01-a Name

FR-03-01-b Email

FR-03-01-c Assigned role

FR-03-01-d Account status (enabled or disabled)

FR-03-01-03 The system shall allow Admins to access a user management interface for a specific user from the user list.

US-03-02 Search, filter, and sort users (Admin)

Priority: 1

Story:

As an Admin, I want to search and filter users so that I can quickly locate a specific user account.

Functional Requirements:

FR-03-02-01 The system shall allow Admins to search for users by name or email.

FR-03-02-02 The system shall allow Admins to filter users by role.

FR-03-02-03 The system shall allow Admins to sort the user list by displayed attributes.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

US-03-03 Manage user account information (Admin)

Priority: 2

Story:

As an Admin, I want to modify a user's account information so that the information remains accurate.

Functional Requirements:

- FR-03-03-01 The system shall allow Admins to modify the user's name.
- FR-03-03-02 The system shall allow Admins to modify the user's email address.
- FR-03-03-03 The system shall validate that updated email addresses remain unique within the system.
- FR-03-03-04 The system shall provide navigation allowing Admins to return to the users list.

US-03-04 Assign user role (Admin)

Priority: 5

Story:

As an Admin, I want to assign roles to users so that they are granted the appropriate level of system access.

Functional Requirements:

- FR-03-04-01 The system shall allow Admins to assign a new role to an existing user.
- FR-03-04-02 The system shall ensure that a user has exactly one assigned role at a time.
- FR-03-04-03 The system shall ensure that a user role change has effect on the user's next data fetch.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

US-03-05 Assign user products (Admin)

Priority: 5

Story:

As an Admin, I want to assign products to users so that users are granted access to the products relevant to their work.

Functional Requirements:

- FR-03-05-01 The system shall allow Admins to assign one or more products to a user.
- FR-03-05-02 The system shall allow Admins to remove product assignments to a user.
- FR-03-05-03 The system shall only allow assignment of products that exist in the system.
- FR-03-05-04 The system shall use assigned products to determine which product-related data a user is authorised to access.

US-03-06 Reset user password (Admin)

Priority: 2

Story:

As an Admin, I want to reset a user's password so that the user can regain access to the system when needed.

Functional Requirements:

- FR-03-06-01 The system shall allow Admins to set a new password for a user account.
- FR-03-06-02 The system shall enforce and display the defined password requirements when an Admin sets a new password.
- FR-03-06-03 After an Admin resets a user's password, the system shall allow the user to authenticate using the new password.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-03-06-04 After an Admin has reset a user's password, the system shall require the user to set a new personal password upon the next successful authentication before further system access is granted.

US-03-07 Disable user account (Admin)

Priority: 1

Story:

As an Admin, I want to disable a user account so that the user is prevented from accessing the system without deleting their account information.

Functional Requirements:

FR-03-07-01 The system shall allow Admins to disable an existing user account.

FR-03-07-02 The system shall reject a disabled user's authentication attempt.

FR-03-07-03 The system shall allow Admins to re-enable a previously disabled user account.

FR-03-07-04 Disabling a user account shall not remove the user's existing data or associations within the system.

EP-04 Product Context Selection

This epic covers the selection of the active product within the system. The selected product determines which data and functionality are available to the user across the application.

US-04-01 Select active product

Priority: 5

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Story:

As a user, I want to select the product I am working with so that all relevant system functionality applies to the correct product.

Functional Requirements:

FR-04-01-01 The system shall allow users to select an active product from the products assigned to them.

FR-04-01-02 The system shall display the currently selected product in the application interface.

FR-04-01-03 The system shall apply the selected product when retrieving or storing data in the system.

US-04-02 Persist selected product

Priority: 5

Story:

As a user, I want the system to remember my selected product so that I do not need to select it repeatedly.

Functional Requirements:

FR-04-02-01 The system shall retain the selected product while the user is authenticated.

FR-04-02-02 The system shall update the active product when the user selects a different product.

EP-05 Problem Report Import

This epic covers the functionality for importing problem reports into the system from structured datasets. Users shall be able to upload a dataset, review the imported data, resolve validation issues, and confirm the import of problem reports

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

associated with the currently selected product. More detailed information regarding the defined dataset format intended to be imported is presented in chapter 7 Excel File Management.

US-05-01 Upload problem report dataset

Priority: 5

Story:

As a user with write access, I want to upload datasets containing problem reports so that they can be imported into the system.

Functional Requirements:

FR-05-01-01 The system shall allow users with write access to upload a dataset containing a problem report.

FR-05-01-02 The system shall support uploading datasets in the defined Excel template format.

FR-05-01-03 The system shall validate the uploaded problem report against the currently selected product.

Related Information:

- 6.1 Excel Template Structure

US-05-02 Map and validate uploaded data

Priority: 5

Story:

As a user with write access, I want the system to display the imported data mapped to the fields in a detailed problem report interface so that I can review the information before finalising the import.

Functional Requirements:

FR-05-02-01 The system shall map imported dataset fields to the corresponding problem report fields in the system.

FR-05-02-02 The system shall display the imported data in a detailed problem report interface.

FR-05-02-03 The system shall indicate validation errors for missing or invalid field values.

FR-05-02-04 The system shall prevent submission of the problem report if validation errors exist.

FR-05-02-05 The system shall validate whether an identical problem report already exists in the system and prevent submission of the imported report if a duplicate is detected.

US-05-03 Confirm import

Priority: 5

Story:

As a user with write access, I want to confirm the import of an uploaded, validated problem report so that it is stored in the system.

Functional Requirements:

FR-05-03-01 The system shall allow users to submit a problem report if all required fields pass validation.

FR-05-03-02 The system shall send the problem report data to the backend for validation and storage.

FR-05-03-03 The system shall automatically generate and append a unique Report ID.

FR-05-03-04 The Report ID shall follow the format: PR-YYYYNNN, where YYYY is the year and NNN is a sequential number.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-05-03-05 The system shall prevent submission while validation errors are present.

FR-05-03-06 The system shall notify the user if the backend rejects the submission.

FR-05-03-07 The system shall notify the user if the import of the problem report is successful.

FR-05-03-08 The system shall, on successful problem report imports, initiate a download of the imported report with its report ID as its file name.

FR-05-03-09 On user input, the system shall reset the import problem report flow.

Related Information:

- 6.2 Excel File Archivation

EP-06 Problem Report Management

This epic covers the functionality for viewing and managing imported problem reports. Users with view access shall be able to browse, search, filter, and inspect imported problem reports.

US-06-01 View list of problem reports

Priority: 4

Story:

As a user with view access, I want to view a list of problem reports so that I can browse and inspect reports associated with the currently selected product.

Functional Requirements:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-06-01-01 The system shall allow users with view access to view a list of problem reports associated with the currently selected product.

FR-06-01-02 The problem report list shall at least display the following information:

FR-06-01-a	Report ID
FR-06-01-b	Unit ID
FR-06-01-c	Failure Date
FR-06-01-d	Breakdown Items
FR-06-01-e	Discipline
FR-06-01-f	Failure Area
FR-06-01-g	Root Cause

FR-06-01-03 The system shall indicate whether a problem report is marked as disabled when displayed.

FR-06-01-04 The system shall allow users with view access to navigate to a detailed problem report interface from the list.

US-06-02 Search, filter, and sort problem reports

Priority: 1

Story:

As a user with view access, I want to search, filter, and sort problem reports so that I can efficiently locate specific reports.

Functional Requirements:

FR-06-02-01 The system shall allow users to search for Report IDs, Unit IDs, and Breakdown Items.

FR-06-02-02 The system shall allow users to filter the problem reports by available attributes.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-06-02-03 The system shall allow users to sort the problem report list by available attributes.

US-06-03 View problem report details

Priority: 4

Story:

As a user with view access, I want to view the details of a problem report so that I can examine the recorded information about the reported problem.

Functional Requirements:

FR-06-03-01 The system shall allow users with view access to view a detailed problem report interface.

FR-06-03-02 The system shall only allow users to access problem reports associated with the currently selected product.

FR-06-03-03 The system shall display all information associated with the selected problem report.

FR-06-03-04 The system shall provide navigation to allow the user to return to the problem report list.

Related Information:

- 6.1.1 Problem Report Form

US-06-04 Disable a problem report

Priority: 2

Story:

As an Admin or Editor, I want to disable a problem report so that incorrectly imported reports can be marked as removed.

Functional Requirements:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-06-04-01 The system shall allow Editors or Admins to disable a problem report.

FR-06-04-02 Disabling a problem report shall mark the report as disabled rather than permanently removing it from the system.

EP-07 RAM Analysis

This epic covers the functionality for analysing collected RAM and failure data to support decision-making and RAM analysis. Users with view access shall be able to filter data based on relevant parameters, perceive calculated key RAM metrics, and visualise results through various charts.

US-07-01 Filters for analysis

Priority: 5

Story:

As an user with view access, I want to filter analysis data based on relevant parameters,
so that I can analyse specific subsets of problem reports relevant to my needs.

Functional Requirements:

FR-07-01-01 The system shall provide a filter component available across applicable analysis interfaces.

FR-07-01-02 The system shall allow the user to filter data based on:

- FR-07-01-a Unit ID
- FR-07-01-b Breakdown Items
- FR-07-01-c Customer
- FR-07-01-d Disciplines
- FR-07-01-e Failure Areas

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-07-01-f No Fault Found

FR-07-01-g Root Cause

FR-07-01-h Date Range

FR-07-01-03 The system shall allow the user to combine multiple filters simultaneously.

FR-07-01-04 The system shall update the analysis results when the user applies the filters.

FR-07-01-05 The system shall display a descriptive text indicating how many problem reports, out of the total, the query resulted in.

FR-07-01-06 The system shall allow the user to clear all filters using a clear filters function.

FR-07-01-07 The system shall ensure that all analysis visualisations and calculations reflect the applied filters.

US-07-02 Visualise failure data distribution

Priority: 5

Story:

As a user with view access, I want to visualise problem report data in pie charts so that I can easily understand the distribution of key categories.

Functional Requirements:

FR-07-02-01 The system shall display pie charts for:

FR-07-02-a No Fault Found

FR-07-02-b Root Cause

FR-07-02-c Disciplines

FR-07-02-d Failure Areas

FR-07-02-02 The system shall calculate and display the percentage distribution for each category within a pie chart.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-07-02-03 The system shall handle cases where no data is available by displaying an appropriate empty state.

US-07-03 Visualise failure occurrence over time

Priority: 5

Story:

As a user with view access, I want to visualise the occurrence of failures over time in a bar chart so that I can identify trends and patterns in failure data.

Functional Requirements:

FR-07-03-01 The system shall display a bar chart showing the number of problem reports per selected time interval.

FR-07-03-02 The system shall allow the user to group data by weeks and months.

FR-07-03-03 The system shall update the chart based on the selected grouping.

FR-07-03-04 The system shall ensure that the x-axis represents time intervals based on the selected grouping.

FR-07-03-05 The system shall ensure that the y-axis represents the number of problem reports.

Related Information:

- 9 RAM Metric Definitions and Calculations

US-07-04 Display calculated RAM metrics

Priority: 5

Story:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

As a user with view access, I want to view calculated RAM-related metrics based on problem report data so that I can evaluate system reliability, availability, and maintainability.

Functional Requirements:

FR-07-04-01 The system shall calculate and display:

FR-07-04-a MTBF (Mean Time Between Failures) in hours

FR-07-04-b MDBF (Mean Distance Between Failures) in
kilometers

FR-07-04-c MTTR (Mean Time To Repair) in hours

FR-07-04-d Availability

FR-07-04-02 The system shall display calculated values in a dedicated
section.

Related Information:

- 9 RAM Metric Definitions and Calculations

US-07-05 Display predicted and targeted RAM metrics

Priority: 4

Story:

As a user with view access, I want to view predicted and targeted RAM metrics for a product so that I can compare them with calculated values and assess system performance.

Functional Requirements:

FR-07-05-01 The system shall display the predicted and targeted RAM
metrics:

FR-07-05-a MTBF

FR-07-05-b MDBF

FR-07-05-c MTTR

FR-07-05-d Availability

FR-07-05-02 The system shall present predicted and targeted values in a dedicated section.

FR-07-05-03 The system shall allow a user with view access to visually compare predicted, targeted, and calculated values side by side.

FR-07-05-04 The system shall always display the predicted and targeted RAM metrics on the product level, for comparison if the applied filters do not have registered values.

US-07-06 Visualise cumulative average trend

Priority: 5

Story:

As a user with view access, I want to visualise cumulative average trends of RAM metrics over time so that I can analyse performance development.

Functional Requirements:

FR-07-06-01 The system shall display a line chart showing cumulative average values over time.

FR-07-06-02 The system shall allow the user with view access to select which RAM metric to visualise (MTBF, MDBF, MTTR, Availability).

FR-07-06-03 The system shall update the chart based on the selected metric.

FR-07-06-04 The system shall ensure that the x-axis represents time and the y-axis represents the selected cumulative metric value.

FR-07-06-05 The system shall plot the predicted and targeted RAM metric values if they are available.

FR-07-06-06 The system shall handle cases where insufficient data is available by displaying an appropriate empty state.

Related Information:

- 9 RAM Metric Definitions and Calculations

US-07-07 Visualise reliability trends

Priority: 5

Story:

As a user with view access, I want to visualise reliability trends so that I can assess reliability growth and system behaviour over time.

Functional Requirements:

FR-07-07-01 The system shall display a Duane plot representing reliability growth over time.

FR-07-07-02 The system shall display a Weibull plot representing reliability characteristics.

FR-07-07-03 The system shall allow a user with view access to select the basis for analysis on either:

FR-07-07-a time

FR-07-07-b distance

FR-07-07-04 The system shall calculate and display the growth rate (α) for the Duane model.

FR-07-07-05 The system shall display the interpreted trend based on the growth rate (positive when $\alpha \geq 0$).

FR-07-07-06 The system shall calculate and display Weibull parameters including shape parameter (β) and characteristic life (η).

FR-07-07-07 The system shall display a goodness-of-fit measure (R^2) for the Weibull and Duane model.

FR-07-07-08 The system shall handle cases where insufficient data is available by displaying an appropriate empty state.

Related Information:

- 9 RAM Metric Definitions and Calculations

US-07-08 Units overview

Priority: 5

Story:

As a user with view access, I want to see a list of all units belonging to a selected product, including their calculated, predicted, and targeted RAM metrics, so that I can get an overview of unit performance.

Functional Requirements:

FR-07-08-01 The system shall allow a user with view access to display a table containing all units associated with the selected product.

FR-07-08-02 The system shall display the following RAM metrics for each unit:

FR-08-01-a MTBF

FR-08-01-b MDBF

FR-08-01-c MTTR

FR-08-01-d Availability

FR-07-08-03 The system shall present the following values side by side for each unit:

FR-08-01-a Calculated RAM metrics

FR-08-01-b Predicted RAM metrics

FR-08-01-c Targeted RAM metrics

FR-07-08-04 If a unit does not have predicted and/or targeted RAM metric values registered, the system shall display the corresponding predicted and targeted values from the product level for comparison.

FR-07-08-05 The system shall indicate whether a calculated value performs worse than its corresponding predicted or targeted value.

EP-08 Action Report Management

This epic covers the functionality for creating, managing, and viewing action reports within the system. The goal is to support traceability, structured documentation, and effective follow-up of decisions and corrective actions.

US-08-01 Create a new action report

Priority: 4

Story:

As an Editor or Admin, I want to create action reports so that decisions and corrective actions related to change requests can be documented and tracked.

Functional Requirements:

FR-07-08-06 The system shall provide Editors or Admins a form for creating a new action report.

FR-07-08-07 The action report form shall include the fields:

- | | |
|------------|--|
| FR-08-01-a | Product |
| FR-08-01-b | Date |
| FR-08-01-c | Decision maker name |
| FR-08-01-d | Decision maker role |
| FR-08-01-e | Change reference |
| FR-08-01-f | Description |
| FR-08-01-g | Corrective actions |
| FR-08-01-h | Replaced Components or Service Actions |
| FR-08-01-i | Comment |

FR-07-08-08 The system shall automatically generate a unique Report ID for each new action report.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-07-08-09 The Report ID shall follow the format: AR-YYYYNNN,
where YYYY is the year and NNN is a sequential number.

FR-07-08-10 The system shall associate the action report with the currently
selected product.

Related Information:

- 7 Action Report Data Definition

US-08-02 View list of action reports

Priority: 4

Story:

As a user with view access, I want to view a list of action reports so that I can
browse and inspect reports associated with the currently selected product.

Functional Requirements:

FR-08-02-01 The system shall allow users with view access to view a list of
action reports associated with the currently selected product.

FR-08-02-02 The action reports list shall at least display the following
information:

FR-08-02-a	Report ID
FR-08-02-b	Date
FR-08-02-c	Decision Maker
FR-08-02-d	Corrective Actions
FR-08-02-e	Change Reference
FR-08-02-f	Created By
FR-08-02-g	Updated By

FR-08-02-03 The system shall indicate whether an action report is marked
as disabled

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-08-02-04 The system shall indicate whether an action report is marked as completed.

FR-08-02-05 The system shall allow users with view access to navigate to the detailed action report interface from the list.

US-08-03 Search, filter, and sort action reports

Priority: 1

Story:

As a user with view access, I want to search, filter, and sort action reports so that I can efficiently locate specific reports.

Functional Requirements:

FR-08-03-01 The system shall allow users to search for action reports by Report ID, Decision Maker, Change Reference, Created By, and Updated By.

FR-08-03-02 The system shall allow users to filter the action report list by available attributes.

FR-08-03-03 The system shall allow users to sort the action report list by available attributes.

US-08-04 View action report details

Priority: 4

Story:

As a user with view access, I want to view detailed information about an action report so that I can inspect decisions, corrective actions, and related data.

Functional Requirements:

FR-08-04-01 The system shall allow users with view access to view a detailed action report interface.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-08-04-02 The system shall ensure that users only can view action reports associated with the currently selected product.

FR-08-04-03 The system shall display all information associated with the selected action report.

FR-08-04-04 The system shall provide navigation to allow the user to return to the action report list.

Related Information:

- 7 Action Report Data Definition

US-08-05 Disable an action report

Priority: 2

Story:

As an Admin or Editor, I want to disable an action report so that incorrect reports can be marked as removed.

Functional Requirements:

FR-08-05-01 The system shall allow Editors or Admins to disable an action report.

FR-08-05-02 Disabling an action report shall mark the report as disabled rather than permanently removing it from the system.

US-08-06 Edit an action report

Priority: 4

Story:

As an Editor or Admin, I want to edit action reports so that change requests which have come further in their process can be documented.

Functional Requirements:

FR-08-06-01 The system shall provide a form for editing action reports.

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

FR-08-06-02 The system shall allow an Editor or Admin to save the action report changes.

FR-08-06-03 The system shall track which user the action report was created and last updated by.

US-08-07 Complete an action report

Priority: 4

Story:

As an Admin or Editor, I want to mark an action report as completed so that no further edits can be done.

Functional Requirements:

FR-08-07-01 The system shall allow Editors or Admins to mark an action report as completed.

FR-08-07-02 The system shall ensure action reports marked as completed cannot be edited.

FR-08-07-03 The system shall save and present the date which marked the action as completed.

EP-09 Export Data

This epic covers the functionality for exporting data from the system in a structured XML format compliant with the S5000F in-service data feedback specification. It includes exporting action reports and calculated RAM metrics to support interoperability, traceability, and external analysis.

US-09-01 Export RAM metrics

Priority: 4

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Story:

As a user with view access, I want to export calculated RAM metrics in S5000F-compliant XML format so that I can use the data for analysis and reporting in external systems.

Functional Requirements:

FR-09-01-01 The system shall provide an option in the RAM analysis interface to export calculated RAM metrics.

FR-09-01-02 The system shall generate an XML file containing the calculated RAM metrics.

FR-09-01-03 The system shall structure the exported data in accordance with the S5000F in-service data feedback specification.

FR-09-01-04 The system shall include the calculated RAM metrics for:

FR-09-01-a MTBF

FR-09-01-b MDBF

FR-09-01-c MTTR

FR-09-01-d Availability

FR-09-01-05 The system shall ensure that exported data reflects the currently applied filters.

FR-09-01-06 The system shall initiate a download of the generated XML file.

Related Information:

- 8.1 RAM Metrics Export

US-09-02 Export action report

Priority: 4

Story:

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

As a user with view access, I want to export an action report in S5000F-compliant XML format so that I can integrate the report with external systems.

Functional Requirements:

FR-09-02-01 The system shall provide an option in the detailed action report interface to export an action report if it is marked as completed.

FR-09-02-02 The system shall generate an XML file containing the action report data.

FR-09-02-03 The system shall structure the exported data in accordance with the S5000F in-service data feedback specification.

FR-09-02-04 The system shall initiate a download of the generated XML file.

Related Information:

- 8.2 Action Report Export

EP-10 Authorisation Warnings

This epic covers the functionality for handling and displaying user feedback when access to the system or specific functionality is restricted due to missing roles, product selection, or insufficient authorisation. The goal is to provide clear guidance to users and prevent unintended system usage.

US-10-01 Display warning for user without assigned role and/or product

Priority: 5

Story:

As a user, I want to be informed when I do not have an assigned role or product, so I understand why I cannot access the system and how to proceed.

Functional Requirements:

- FR-10-01-01 The system shall detect when a user has no assigned role and/or product.
- FR-10-01-02 The system shall display a warning message informing the user that they do not have an assigned role and/or product.
- FR-10-01-03 The system shall instruct the user to contact the system administrator.
- FR-10-01-04 The system shall prevent access to system functionality until a role and/or product has been assigned.

US-10-02 Display warning for user without a selected product

Priority: 5

Story:

As a user, I want to be informed when I have not selected a product, so that I know I must choose one before accessing functionality.

Functional Requirements:

- FR-10-02-01 The system shall detect when a user attempts to access functionality without having a product selected.
- FR-10-02-02 The system shall display a warning message informing the user that no product has been selected.
- FR-10-02-03 The system shall instruct the user to select a product to proceed.
- FR-10-02-04 The system shall prevent access to product-specific functionality until a product is selected.

US-10-03 Display warning for unauthorised access

Priority: 5

Story:

As a user, I want to be informed when I attempt to access functionality I am not authorised to, so that I understand why I am restricted from specific system functionality.

Functional Requirements:

FR-10-03-01 The system shall detect when a user attempts to access a page or product they are not authorised to access.

FR-10-03-02 The system shall display a warning message informing the user that they are not authorised.

FR-10-03-03 The system shall instruct the user to contact the system administrator.

FR-10-03-04 The system shall prevent access to the requested page or functionality.

4 Non-Functional Requirements

This chapter presents the system's non-functional requirements, categorised on security, compliance, performance, usability, maintainability, and availability. Each non-functional requirement is assigned a unique identifier using the format NFR-XXX-YY, where XXX denotes the category, and YY is a sequential number within that category.

4.1 Security

NFR-SEC-01

Passwords shall be stored using a secure hashing algorithm.

NFR-SEC-02

Authenticated sessions shall expire after a defined period of inactivity.

NFR-SEC-03

The system shall invalidate session tokens upon logout.

4.2 Compliance

NFR-COM-01

The system shall generate exported data in XML format compliant with the S5000F in-service data feedback specification.

NFR-COM-02

The system shall ensure that each exported XML message contains a unique message identifier in accordance with S5000F requirements.

NFR-COM-03

The system shall ensure that all generated XML files conform to the defined XSD and are structurally valid.

4.3 Performance

NFR-PER-01

The system shall respond to user interactions within 2 seconds under normal operating conditions.

NFR-PER-02

The system shall update analysis visualisations within 3 seconds after applying filters.

NFR-PER-03

The system shall support concurrent usage by multiple users without significant reduction in performance.

4.4 Usability

NFR-USA-01

The system shall provide a consistent and intuitive user interface across all views.

NFR-USA-02

The system shall provide clear feedback to users when actions are performed (e.g. success messages and error messages).

NFR-USA-03

The system shall validate user input and provide descriptive error messages if validation fails.

4.5 Maintainability

NFR-MAI-01

The system shall be sufficiently modular for future extension and modification of functionality.

NFR-MAI-02

The system shall separate data handling, business logic, and presentation layers.

NFR-MAI-03

The system shall be designed to allow addition of new system configuration without significant changes to existing functionality.

4.6 Availability

NFR-AVA-01

The system shall handle unexpected errors gracefully without crashing.

5 User Interface Concepts

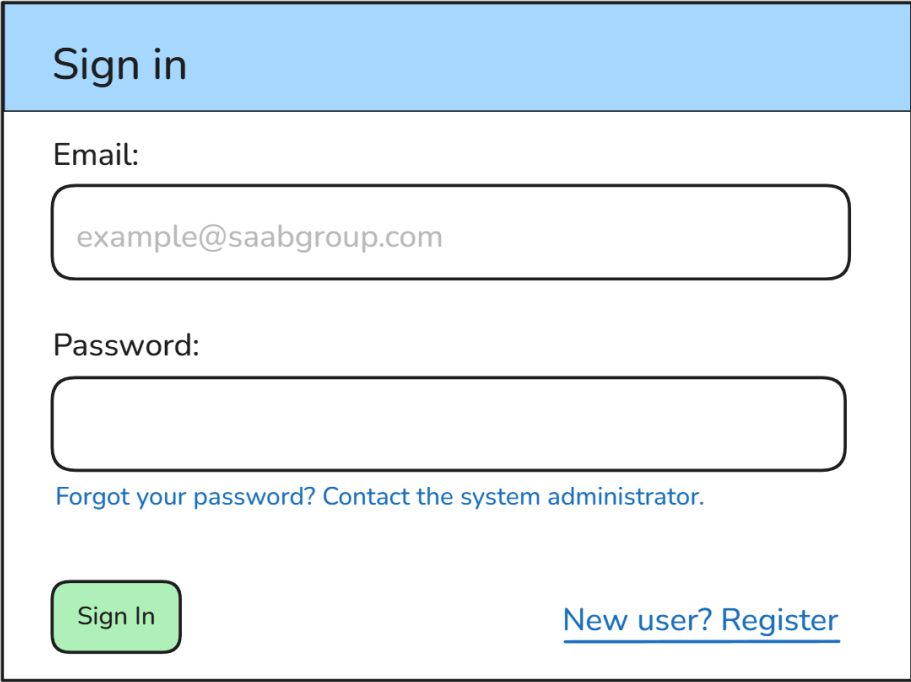
This chapter presents design prototypes made to illustrate the user interface of the application.

5.1 Visitor Interfaces

This section presents interfaces for unauthenticated users.

5.1.1 Sign In

Figure 5.1 presents an interface that allows an unauthenticated user to sign in or navigate to the register form.



The image shows a design prototype for a 'Sign in' interface. It features a light blue header bar with the text 'Sign in'. Below the header, there are two input fields: one for 'Email:' with the placeholder text 'example@saabgroup.com', and one for 'Password:'. Below the password field, there is a link that reads 'Forgot your password? Contact the system administrator.' At the bottom left, there is a green button labeled 'Sign In'. At the bottom right, there is a link that reads 'New user? Register'.

Figure 5.1: An interface that allows an unauthenticated user to sign in or navigate to the register form.

5.1.2 Register

Figure 5.2 presents an interface that allows an unauthenticated user to register or navigate to the sign in form.

The form is titled "Register" in a light blue header. It contains four input fields: "Email:" with the placeholder "example@saabgroup.com", "Full name:" with the placeholder "John Doe", "New Password:", and "Confirm Password:". At the bottom left is a green "Register" button, and at the bottom right is a blue link "Already have an account? Sign in".

Register	
Email:	example@saabgroup.com
Full name:	John Doe
New Password:	
Confirm Password:	
Register	Already have an account? Sign in

Figure 5.2: An interface that allows an unauthenticated user to register or navigate to the sign in form.

5.2 General User Interfaces

This section presents user interfaces that are general, i.e. not role specific.

5.2.1 Landing Page

Figure 5.3 presents an interface that acts as the mutual landing page for all authenticated users, which have an assigned role. Here they are able to select which product they want to work with.

Welcome to the RAM tool, Oscar Cedell

A rectangular button with rounded corners. Inside the button, the word "Select" is centered in a light gray font. To the right of the text is a circular icon containing a stylized letter "V".

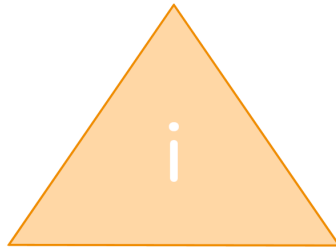
Figure 5.3: An interface that acts as the mutual landing page for all authenticated users.

5.3 Authorisation Warnings

This section presents user interfaces corresponding to authorisation warnings.

5.3.1 Missing Role/Product Warning

Figure 5.4 presents an interface that applies to newly registered users, where the admin has not yet assigned any role and/or products to the user.

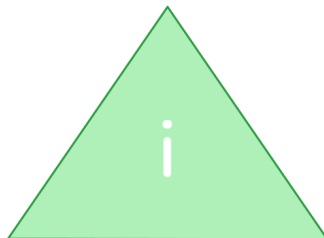


You do not have an assigned role and/or product
Contact the system administrator

Figure 5.4: An interface that applies to newly registered users, where the admin has not yet assigned any role and/or products to the user.

5.3.2 Missing Product Selection Warning

Figure 5.5 presents an interface that applies to a user who tries to access a page which they have authorisation to, but have not selected a product yet.

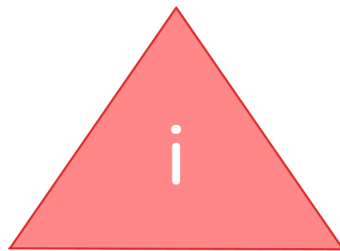


You have not selected a product
Please select a product to access the functionality

Figure 5.5: An interface that applies to a user who tries to access a page which they have authorisation to, but have not chosen a product yet.

5.3.3 Unauthorised Access Warning

Figure 5.6 presents an interface that applies to users who try to access a page or product which they are not authorised to. This warning will not appear through intended normal system usage.



You are not authorised for this page and/or product
Contact the system administrator

Figure 5.6: An interface that applies to users who try to access a page or product which they are not authorised to.

5.4 Admin Interfaces

This section presents user interfaces that are limited to users with the Admin role.

5.4.1 Top Navigation

Figure 5.7 presents an interface that acts as the top navigation for Admins, allowing navigation to all system functionality. It includes dropdowns for easy-access product selection, analysis interfaces, and configuration interfaces.

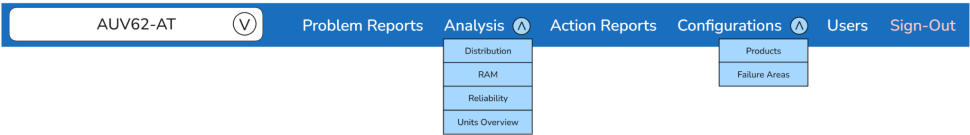


Figure 5.7: An interface that acts as the top navigation for Admins.

5.4.2 Product Management

Figure 5.8 presents an interface that allows an Admin to manage products within the system and navigate to product-specific management interfaces.

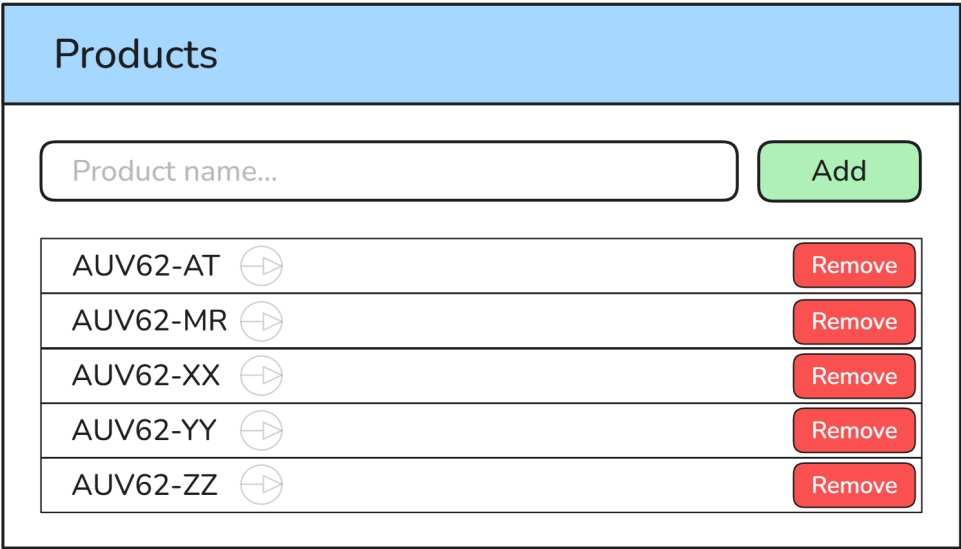


Figure 5.8: An interface that allows an Admin to manage products within the system and navigate to product-specific management interfaces.

5.4.3 Breakdown Item Management

Figure 5.9 presents an interface that allows an Admin to manage breakdown items assigned to a specific product.

Breakdown items for AUV62-AT

Breakdown item LCN...

Breakdown item name...

Add

A		Nose	Remove
B		Propeller	Remove
C		Mid	Remove
D		Rear	Remove

Figure 5.9: An interface that allows an Admin to manage breakdown items assigned to a specific product.

5.4.4 Unit Management

Figure 5.10 presents an interface that allows an Admin to manage the units for a specific product.

Units for AUV62-AT

Unit name...

Add

S1-X		Remove
S1-Y		Remove
S1-Z		Remove
S2-X		Remove
S2-Y		Remove

Figure 5.10: An interface that allows an Admin to manage the units for a specific product.

5.4.5 Product Predictions Management

Figure 5.11 presents an interface that allows an Admin to manage RAM-related metric predictions for a specific product, unit, or breakdown item.

Predictions for AUV62-AT

Availability:

100

MTTR (h):

8.0

MTBF (h):

20

MDBF (km):

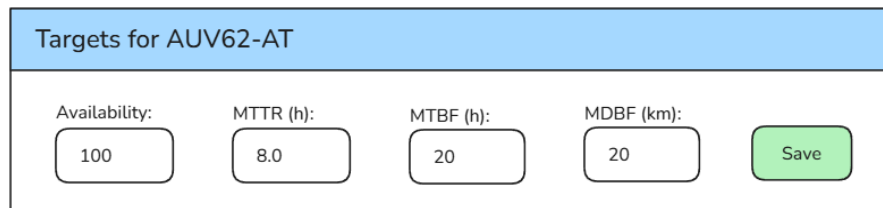
20

Save

Figure 5.11: An interface that allows an Admin to manage RAM-related metric predictions for a specific product, unit, or breakdown item.

5.4.6 Target Management

Figure 5.12 presents an interface that allows an Admin to manage the RAM-related metric targets for a specific product, unit, or breakdown item.

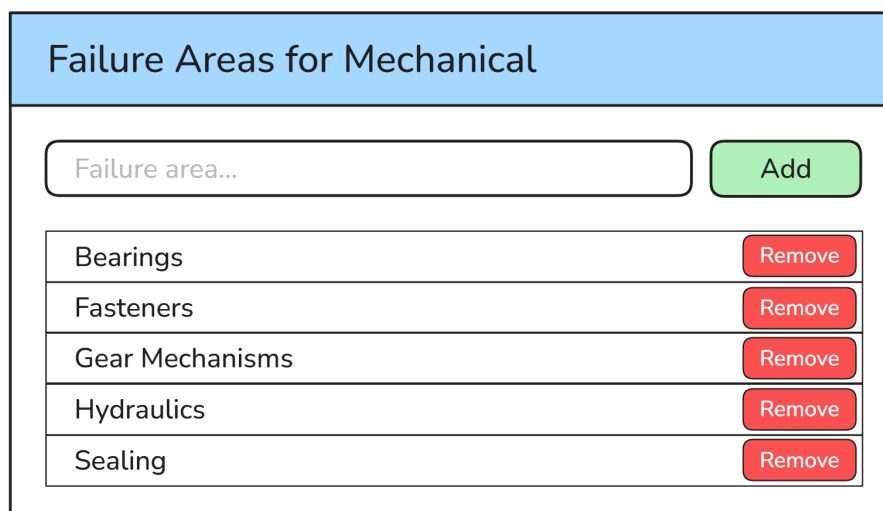


Targets for AUV62-AT				
Availability:	MTTR (h):	MTBF (h):	MDBF (km):	
100	8.0	20	20	Save

Figure 5.12: An interface that allows an Admin to manage the RAM-related metric targets for a specific product, unit, or breakdown item.

5.4.7 Failure Area Management

Figure 5.13 presents an interface that allows an Admin to manage the discipline-specific failure areas available to filter analysis on and validate problem report import against.



Failure Areas for Mechanical	
Failure area...	Add
Bearings	Remove
Fasteners	Remove
Gear Mechanisms	Remove
Hydraulics	Remove
Sealing	Remove

Figure 5.13: An interface that allows an Admin to manage the discipline-specific failure areas available to filter analysis on and validate problem report import against.

5.4.8 User List

Figure 5.14 presents an interface that allows an Admin to get an overview of all users in the system. It highlights users marked as disabled and enables navigation to the detailed user interface.

Users		
Name	Email	Role
Search for a name...	Search for an email...	Select a role 
Oscar Cedell 	oscar.cedell@saabgroup.com	Admin
Jacob Persson 	jacob.persson@saabgroup.com	Viewer
Disabled User 	disabled.user@saabgroup.com	Reporter
John Doe 	john.doe@saabgroup.com	Editor

Figure 5.14: An interface that allows an Admin to get an overview of all users in the system.

5.4.9 User Management

Figure 5.15 presents an interface that allows an Admin to manage a specific user’s account information, assign roles and products, reset password, and disable the user’s access to the system.

User

Full name:

Oscar Cedell

Email:

oscar.cedell@saabgroup.com

Role:

Admin

Disabled:

No

Assigned Products:

Choose a product

Add

AUV62-AT	Remove
AUV62-MR	Remove
AUV62-XX	Remove

Close

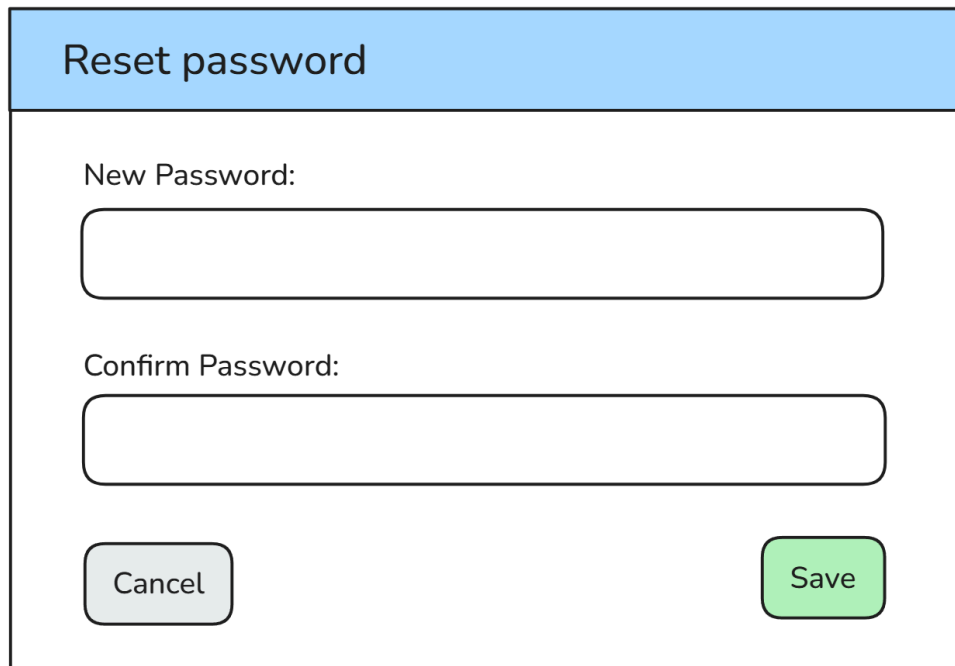
Reset Password

Save

Figure 5.15: An interface that allows an Admin to manage a user's information and access.

5.4.10 Reset Password

Figure 5.16 presents an interface that allows an Admin to reset a user's password.



The interface is a modal window titled "Reset password" in a blue header bar. Below the header, there are two text input fields. The first is labeled "New Password:" and the second is labeled "Confirm Password:". At the bottom of the modal, there are two buttons: a light gray "Cancel" button on the left and a green "Save" button on the right.

Figure 5.16: An interface that allows an Admin to reset a user's password.

5.5 Reporter Interfaces

This section presents user interfaces that are designated to users with the Reporter role.

5.5.1 Top Navigation

Figure 5.17 presents an interface that acts as the top navigation component for users with the Report role. Reporters can only select the product they are currently working with or sign out.



Figure 5.17: An interface that acts as the top navigation component for users with the Report role.

5.5.2 Start Import Problem Report Flow

Figure 5.18 presents an interface that acts as the start of the import problem report flow. It is a button allowing a user with write access to upload a problem report from an Excel file.

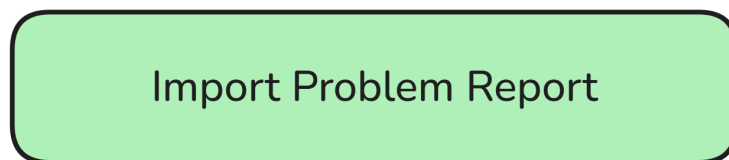


Figure 5.18: An interface that acts as the start of the import problem report flow.

5.5.3 Display Problem Report Validation Errors

Figure 5.19 presents an interface that is presented to a user with write access when they have imported a problem report that contains validation errors. The submit button is disabled and the user is allowed to navigate back to the start of the import problem report flow. “...” indicates that fields have been omitted from the interface concept.

Validate Uploaded Problem Report

Product:

AUV62-MR

* Product value does not match selected product

Unit ID:

AABBCCDDEE

Customer:

* Customer value missing

Reporter:

Problem Report Creator

...

Cancel

Submit

Figure 5.19: An interface that indicates that an imported problem report contains validation errors.

5.5.4 Inspect Uploaded Problem Report

Figure 5.20 presents an interface that is presented to a user with write access when they have imported a problem report that does not contain any validation errors. The submit button is enabled and the user may submit the problem report to the system. The user is also presented with an option to navigate back to the start of the import problem report flow. “...” indicates that fields have been omitted from the interface concept.

Validate Uploaded Problem Report

Product:

AUV62-AT

Unit ID:

AABBCCDDEE

Customer:

Example Customer

Reporter:

Problem Report Creator

...

Cancel

Submit

Figure 5.20: An interface indicating that an imported problem report does not contain any validation errors

5.5.5 Display Problem Report Import Error

Figure 5.21 presents an interface that is shown when a user submits a problem report and the backend service rejects the submission. The interface presents the user with an error message and is allowed to navigate back to the start of the import problem report flow.

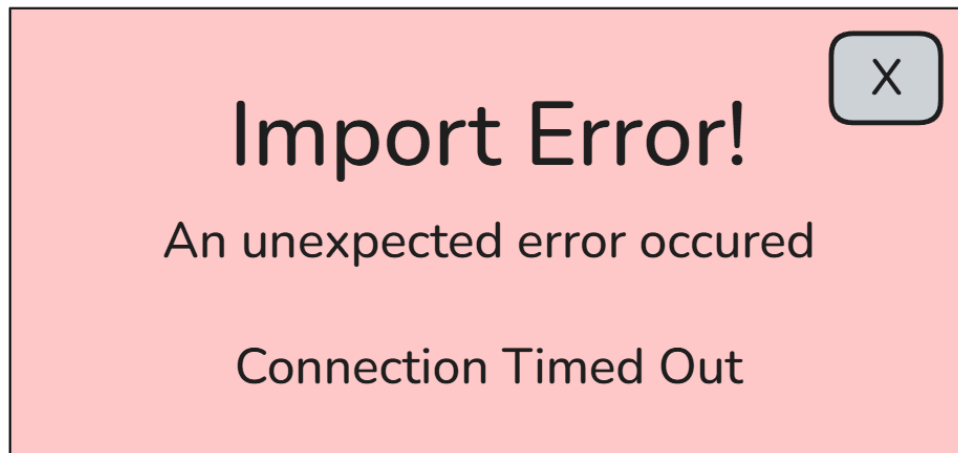


Figure 5.21: An interface that is shown when a user submits a problem report and the backend service rejects the submission.

5.5.6 Display Problem Report Import Confirmation

Figure 5.22 presents an interface that is shown when a user submits a problem report and the backend service accepts the submission. The interface presents the user with the inserted problem report ID and a prompt to handle the excel file for the imported problem report. For description of the storage process see [6.2 Excel File Archivation](#).

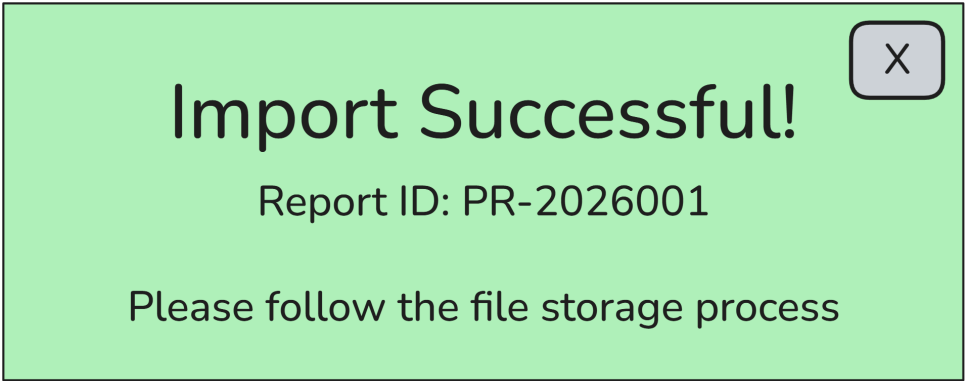


Figure 5.22: An interface that is shown when a user submits a problem report and the backend service accepts the submission.

5.6 Analyst Interfaces

This section presents user interfaces that are designated to users with the Analyst role.

5.6.1 Top Navigation

Figure 5.23 presents an interface that acts as the top navigation for Analysts. It allows navigation to all functionality an analyst has access to. It includes dropdowns for easy-access product selection and analysis interfaces.

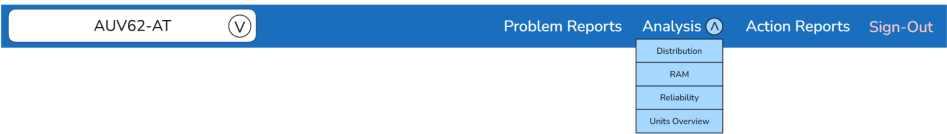


Figure 5.23: An interface that acts as the top navigation for Analysts.

5.6.2 Problem Report List

Figure 5.24 presents an interface that lists all problem reports associated with the selected product. It allows navigation to the detailed problem report interface. An Editor or Admin may also import a new problem report from this view. Reports marked as disabled are highlighted.

Problem Reports						
Import Problem Report						
Report ID	Unit ID	Failure Date	Breakdown Item	Discipline	Failure Area	Root Cause
Search for a report ID...	Search for a unit ID...	YYYY-MM-DD - YYYY-MM-DD	Search for an item...	Select a discipline	Select a failure area	Select a root cause
PR-2026030	AABBCCDDEE	2026-03-30	A - Nose	Mechanical	Sealing	Operator Error
PR-2026025	AABBCCDDEE	2026-03-25	B - Propeller	Software	Area 1	Component Error
PR-2026020	EEDDCCBBAA	2026-02-22	C - Front	Software	Area 2	Component Error
PR-2026018	XXYYZZYYXX	2026-02-20	D - Rear	Electrical	Power	Operator Error
Deleted Report	WWXXYYZZ	2026-01-21	E - Mid	Mechanical	Bearings	Operator Error

Figure 5.24: An interface that lists all problem reports associated with the selected product.

5.6.3 Detailed Problem Report

Figure 5.25 presents an interface that displays all information regarding a specific problem report. It also allows navigation back to the problem report list and Editors or Admins to mark the problem report as disabled. “...” indicates that fields have been omitted from the interface concept.

Problem Report

Report ID:

PR-2026001

Product:

AUV62-AT

Unit ID:

AABBCCDDEE

Customer:

Example Customer

Reporter:

Problem Report Creator

...

Cancel

Figure 5.25: An interface that displays all information regarding a specific problem report.

5.6.4 Analysis Filter Component

Figure 5.26 presents a component that is a filter which is mutual for all applicable analysis interfaces. It includes a description of the resulting number of failure reports that have been filtered, as well as the total number of problem reports associated with the selected product. The component also includes a clear filter button for easy resetting of the filters.

Filters

Unit ID:

Search for a unit ID...

Breakdown Item

Select modules

Customer:

Search for a customer...

Disciplines:

Select disciplines

Failure Areas:

Select failure areas

No Faults Found:

Select Yes/No

Root Cause:

Select root cause

Date:

YYYY-MM-DD

YYYY-MM-DD

Search

Clear Filters

Query resulted in 512 Problem Reports out of 512 total.

Figure 5.26: A component acting as a mutual filter for all applicable analysis interfaces.

5.6.5 Failure Data Distribution Visualisation

Figure 5.27 presents an interface that displays pie charts of problem report distributions of different categories. Each pie chart includes a legend describing its values.

194

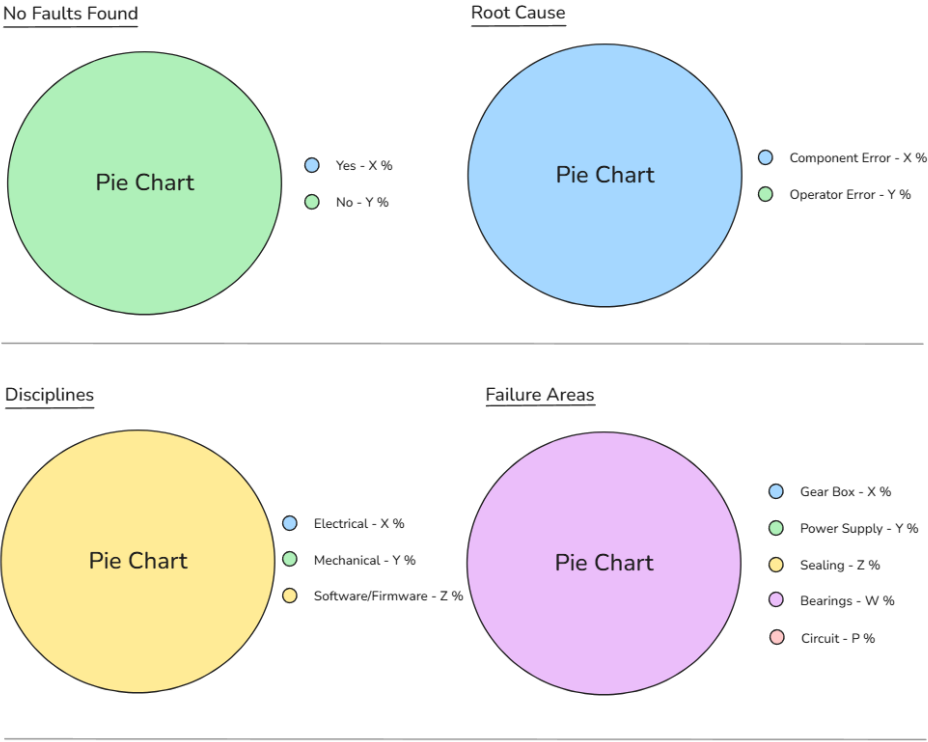


Figure 5.27: An interface that displays pie charts of problem report distributions of different categories.

5.6.6 Failure Occurrence Trend

Figure 5.28 presents an interface that allows a user to analyse the failure occurrence trend for the selected product. The user may group the problem reports in weeks or months.

Failure Occurrence Trend

Group Reports By:

Weeks

Months

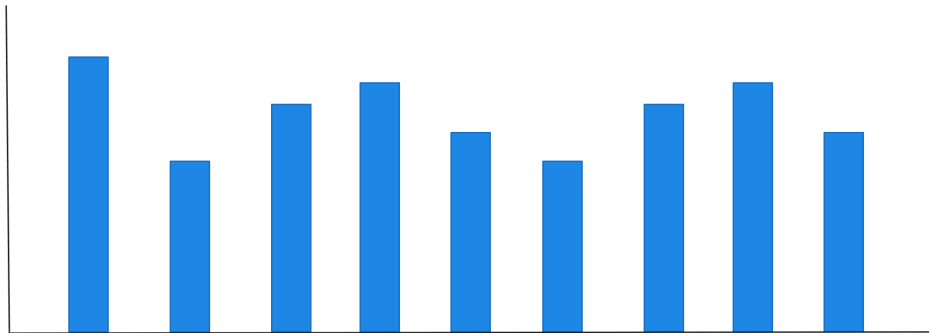


Figure 5.28: An interface that allows a user to analyse the failure occurrence trend for the selected product.

5.6.7 RAM Metrics Analysis

Figure 5.29 presents an interface that displays the predictions and targets available for the selected product. The interface also presents calculated values, based on the filters. The user may export these calculated values to XML through a button. The calculated values are also visualised in a plot which depicts their cumulative average based on the metric selection. Predictions and targets are also visualised in the plot if selected.

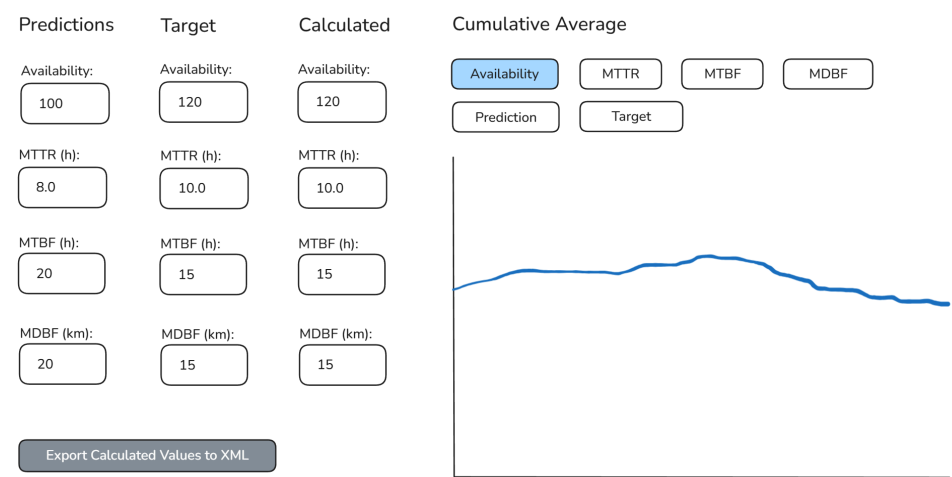


Figure 5.29: An interface that displays the predicted, targeted, and calculated RAM metrics.

5.6.8 Reliability Trends

Figure 5.30 presents an interface that displays a Duane and Weibull plot, with the possibility to base the plot on either time or distance. It also presents respective relevant values, such as growth rate and shape parameters.

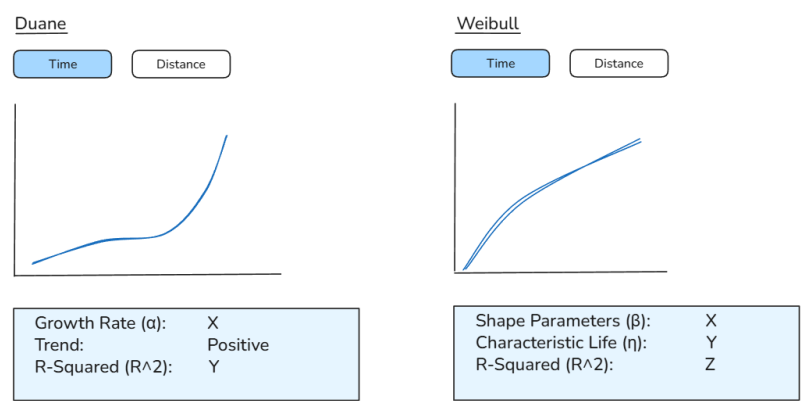


Figure 5.30 An interface that displays a Duane and Weibull plot, with the possibility to base the plot on either time or distance.

5.6.9 Action Report List

Figure 5.31 presents an interface that lists all action reports associated with the selected product. Reports marked as disabled are highlighted. The list allows a user to navigate to the detailed action report interface. The interface also allows Editors or Admins to add new action reports.

Action Reports

New Action Report

Report ID	Date	Decision Maker	Corrective Actions	Change Reference	Created By	Updated By
Search for an ID...	YYYY-MM-DD - YYYY-MM-DD	Search for a name...	Select an action	Search for a reference...	Search for an email...	Search for an email...
AR-2026016	2026-03-30	Oscar Cedell	Design Change	CCB-XXYYZZ	oscar.cedell@saabgroup.com	oscar.cedell@saabgroup.com
AR-2026012	2026-03-25	Oscar Cedell	Educational Effort	MoM-112233	oscar.cedell@saabgroup.com	oscar.cedell@saabgroup.com
AR-2026002	2026-02-22	Jacob Persson	Technical Publications	CO-123123	oscar.cedell@saabgroup.com	oscar.cedell@saabgroup.com
AR-2025098	2025-12-20	Oscar Cedell	Educational Effort	MoM-XXYYZZ	oscar.cedell@saabgroup.com	oscar.cedell@saabgroup.com
Deleted Report	2025-01-21	Jacob Persson	Design Change	CO-998877	oscar.cedell@saabgroup.com	oscar.cedell@saabgroup.com

Figure 5.31: An interface that lists all action reports associated with the selected product.

5.6.10 New Action Report Form

Figure 5.32 presents an interface that acts as the form for creating new action reports. The report id and product fields are disabled and filled out automatically by the system. The action report is associated with the currently selected product. The form allows for navigation back to the action report list interface as well as submission.

New Action Report

Report ID:

Auto generated

Date:

2026-03-15

Decision Maker Name:

John Doe

Description:

Product:

AUV62-AT

Change Reference:

MoM for CCB, CO, or similar

Decision Maker Role:

Design Authority

Corrective Actions:

Select Corrective Actions

Replaced Components or Service Actions:

Close

Submit

Figure 5.32: An interface that acts as the form for creating new action reports.

5.6.11 Detailed Action Report

Figure 5.33 presents an interface that acts as the form that displays all information regarding a specific action report. It allows a user to navigate back to the action report list interface, mark the action report as disabled or mark the action report as completed.

Action Report

Created: 2026-04-01

By: jacob.persson@saabgroup.com

Updated: 2026-04-03

By: oscar.cedell@saabgroup.com

Completed: -

Report ID:

AR-2026016

Product:

AUV62-AT

Date:

2026-03-30

Change Reference:

CCB-XXYYZZ

Decision Maker Name:

Oscar Cedell

Decision Maker Name:

Design Authority

Corrective Actions:

Design Change

Description:

Propeller 123 and Light 54 have multiple reported failures. Accelerometer needs calibration.

Replaced Components or Service Actions:

Propeller 123 replaced for Propeller 456,
Light 54 replaced for Light 6-7
Accelerometer calibrated

Close

Complete

Disable

Figure 5.33: An interface that acts as the form that displays all information regarding a specific action report.

6 Excel File Management

This chapter presents the work flow for the management of excel files for problem reports as intended input data to the system. It describes the excel template used for collection of problem reports and the management of imported excel files.

200

6.1 Excel File Structure

This section describes the intended excel template for collecting problem reports. The included subsections correspond to individual tabs (i.e. sheets) in the excel file. The first sheet constitutes the problem report form. The second sheet acts as a revision history of the problem report. Finally, the third sheet includes the options needed for fields in the problem report where an option should be selected, separating administrative changes from the form.

6.1.1 Problem Report Form

This sheet constitutes the problem report form. Each input field is supported by an input message (tool-tip) providing a description, data type, requirement status, and an example value. Additionally, each field features an error alert that is triggered if the input does not meet the defined conditions. Table 6.1 provides a structured overview of all input fields, including their respective data types and conditional requirements.

Table 6.1: Overview of input fields, including data types and conditions.

Section	Field	Input Type	Input Condition
Product Information	Product	Dropdown	-
Product Information	Unit ID	Text	-
Product Information	Customer	Text	-
Failure Information	Reporter Email	Text	-
Failure Information	Failure Date	Date	-
Failure Information	Run Time Before Failure	Number	-

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Failure Information	Distance Before Failure	Number	-
Failure Information	Customer Return Date	Date	-
Failure Information	Labor Hours	Number	-
Failure Information	Failure Symptoms	Text	-
Failure Information	Troubleshooting Actions	Text	-
Root Cause Analysis	No Fault Found	Boolean	-
Root Cause Analysis	Root Cause	Dropdown	NFF = No
Root Cause Analysis	Operator Error Type	Text	NFF = No & Root Cause = Operator Error
Root Cause Analysis	Component LCN	Text	NFF = No & Root Cause = Component Error
Root Cause Analysis	Component Part Number	Text	NFF = No & Root Cause = Component Error
Root Cause Analysis	Component Revision (PDM)	Text	NFF = No & Root Cause = Component Error
Root Cause Analysis	Discipline	Dropdown	NFF = No & Root Cause = Component Error
Root Cause Analysis	Failure Area	Dropdown	NFF = No & Root Cause = Component Error

Responsible: Oscar Cedell

Document Number: MTRAM2601

Date: 2026-04-15

Version: 2.0

Root Cause Analysis	Failure Mode	Text	NFF = No & Root Cause = Component Error
Root Cause Analysis	Corrective Actions	Text	NFF = No & Root Cause = Component Error
Root Cause Analysis	Comment	Text	

6.1.2 Revision History

This sheet constitutes the revision history of the problem report. The table includes dates, authors, and implemented changes. An overview of the sheet is provided in Table 6.2.

Table 6.2: Overview of document revisions with two example rows.

Revision	Date	Author	Change
1	2026-02-10	Oscar Cedell	Initial Report
2	2026-02-12	Jacob Persson	Changed Failure Area to Bearings
3			
4			
5			
6			
7			
8			
9			
10			

6.1.3 Reference Data

This sheet contains the values used to populate options in the problem report form. It serves as the source for the dropdown menus. The sheet can be updated over time by an administrator to add or modify available options. To protect the reference data from unintended changes, the sheet is hidden and password-protected, with access limited to authorised administrators.

6.2 Excel File Archivation

This section presents the workflow for storing imported problem reports in a traceable manner.

When a problem report is submitted and accepted in the digital RAM tool, a copy of the corresponding Excel file is downloaded to the user's local machine with its file name set to the automatically generated unique Report ID. This ensures consistent naming and traceability between the stored file and the Report ID registered in the system.

To ensure structured and accessible storage, users are instructed to archive the downloaded file within a predefined folder structure. The root folder is named "Problem Reports" and contains subfolders for each product, which in turn contains subfolders for units. This provides a scalable approach to organise reports over time, allowing efficient retrieval based on product, unit, and problem report ID.

The folder structure is implemented on a designated network drive, accessible to multiple authorised users within the organisation. This enables centralised storage of all problem reports, while access rights can be managed to ensure appropriate

data security. By using a network drive rather than local storage, the risk of data loss is reduced and consistency in file management is maintained across users.

The archived Excel files primarily serve as a traceability and backup mechanism, ensuring that original input data can be reviewed if needed. The digital tool itself remains the primary system for data storage, analysis, and retrieval, while the Excel archive provides an additional layer of documentation and auditability.

7 Action Report Data Definition

This chapter defines the data structure of action reports used within the system. It complements the functional requirements by specifying the meaning, format, and constraints of each data field, presented in Table 7.1.

Table 7.1: Action Report fields' meaning, format, and constraints.

Field	Description	Type	Constraints
Report ID	Unique identifier for the action report	Text	Auto-generated, format AR-YYYYNNN
Product	Product to which the report belongs	Text	Must exist in system, derived from selected product
Date	Date of the decision/action	Date	Required
Decision Maker Name	Name of the decision maker	Text	Required
Decision Maker Role	Role of the decision maker	Text	Required

Change Reference	Reference to change (e.g. MoM of CCB, CO)	Text	Optional
Description	Description of issue, context, and/or reason for the action report	Text	Required
Corrective Actions	Type of corrective action taken	Dropdown	Must be from predefined list
Replaced Components or Service Actions	Description of performed actions	Text	Optional
Comment	Additional notes	Text	Optional
Created By	Email of the user that created the report	Text	Auto-generated
Updated By	Email of the user that last updated the report	Text	Auto-generated
Completed	The date indicating when a user marked the report as completed	Date	Auto-generated

The predefined list for population of the Corrective Actions field:

- Design Change
- Educational Effort
- Technical Publications

8 S5000F Data Export Design

This chapter describes how calculated RAM metrics and completed action reports are exported in compliance with S5000F. It presents the selected exchange use

cases and explains how application data is mapped to the corresponding S5000F structures.

8.1 RAM Metrics Export

Calculated RAM metrics are exported using the S5000F use case *UC50304: Report Availability*. The use case provides structured exchange of reliability and availability related performance information of a serialised product, fulfilling the goal of exporting calculated RAM metrics.

The export uses three templates corresponding to the system segmentation of metrics calculation during analysis: Product, Unit, and Breakdown Item. Each segment represents different levels of the product structure and therefore demands different information contexts. Where applicable, the segmentation also makes it possible to include multiple sets of calculated metrics within the same message.

Each exported metric is mapped to the S5000F class reportable metric. Each reportable metric includes identification of the metric it represents, such as MTBF, MDBF, MTTR, or Availability. This ensures that receiving systems can correctly interpret the meaning of each exported value.

The reportable metrics are nested differently depending on the segmentation generated by the analysis scope. At the product level, reportable metrics are associated with a product variant structure. At the unit level, reportable metrics are associated with a serialised product variant representing the individual unit. These structures allow the exported metrics to be grouped with the configuration context to which they apply.

At the breakdown item level, reportable metrics are instead placed at message level and reference the corresponding breakdown item through its LCN. This approach is used because breakdown items do not support nesting of reportable metrics in the same way as product and serialised variant structures. Since each metric references its corresponding breakdown item, a consistent method for associating calculated values with the relevant breakdown item is provided.

The metric MDBF is not included in the standard S5000F enumeration of performance parameter types. To support the exchange of all calculated metrics, the value MDBF was added as an extension to the valid values included in the schema. Systems importing RAM metric exports are therefore expected to include this parameter type in their schema definition in order to successfully validate the S5000F message.

8.2 Action Report Export

Action reports are exported using the S5000F use case *UC50302: Influencing future design*, which falls under the reliability category. The use case is described as a message which provides necessary information for raising a change request. Hence, the selected use case aligns with the role of action reports within the system as formal documentation of identified problem patterns based on analysis of imported problem reports.

The exported dataset corresponds to the information required to represent an action report as a structured S5000F exchange message supporting change related decision making. The export includes message level metadata, report content, product context, decision responsibility, corrective actions, and supporting

comments mapped to the corresponding S5000F data classes. The top-level mappings are presented in the following list:

- **Message:**
 - **ID, Date, Language, Status,** and **Type** mapped to message
- **Report:**
 - **ID, Description,** and **Date** mapped to document
- **Product** mapped to product
- **Decision Name** and **Role** mapped to person
- **Change Reference** mapped to change request
- **Corrective Actions** mapped to actions
- **Replaced Components or Service Actions** mapped to comment
- **Comment** mapped to comment

Where the S5000F schema does not provide dedicated attributes for certain information, such as decision maker role or free text change references, the remark class is used to be able to export this information while remaining compliant with the schema.

9 Analysis Calculations

This chapter covers the mathematical formulations and analytical methods used for calculating RAM-related metrics within the system. It provides a formal description of how key values such as MTBF, MDBF, MTTR, Availability, and other derived measures are computed based on collected data. The chapter further specifies the methods used to process and analyse data, including calculating cumulative averages, as well as reliability modelling techniques such as Weibull

and Duane analysis. Where applicable, assumptions and limitations of each method are also described.

9.1 Aggregation of Filters

The applicable analysis pages have a mutual filter applied when fetching problem reports to base the analysis on. The overall filtering is straightforward, except for breakdown items. When a breakdown item is selected, all problem reports associated with that item and all of its descendants are included in the response.

9.2 RAM Metrics

Pseudocode for calculation of RAM metrics.

9.2.1 MTBF, MDBF, MTTR and Availability

INPUT:

- problem_reports
- selected_product
- filters

PROCESS:

1. Fetch all problem reports associated with the selected product and all activated filters
2. Determine calculation segments:
 - a. If one or more units are selected:
 - i. create one segment per unit
 - b. Else if one or more breakdown items are selected and no units are selected:
 - i. create one segment per breakdown item

- c. Else:
 - i. create one segment containing all reports matching the filters
- 3. For each segment:
 - a. For each segment select all reports associated with that segment
 - b. Calculate MTBF and MDBF:
 - i. Sort the segment's reports by Unit and Failure Date in ascending order
 - ii. For each Unit, calculate time and distance between failures using consecutive differences, where the first entry is only used as a baseline:
 - 1. $\text{delta_time_j} = \text{TimeToFailure_j} - \text{TimeToFailure_}(j-1)$
 - 2. $\text{delta_distance_j} = \text{DistanceToFailure_j} - \text{DistanceToFailure_}(j-1)$
 - iii. For each Unit:
 - 1. $\text{unit_interval_count} = \text{number_of_reports} - 1$
 - 2. $\text{unit_total_time} = \text{sum of all delta_time values}$
 - 3. $\text{unit_total_distance} = \text{sum of all delta_distance values}$
 - 4. If $\text{unit_interval_count} > 0$:
 - a. $\text{unit_MTBF} = \text{unit_total_time} / \text{unit_interval_count}$
 - b. $\text{unit_MDBF} = \text{unit_total_distance} / \text{unit_interval_count}$
 - 5. If segments not on Unit level, aggregate MTBF and MDBF across units using weighted averages:

- a. $MTBF = \frac{\sum(\text{unit_MTBF} * \text{unit_interval_count})}{\sum(\text{unit_interval_count})}$
- b. $MDBF = \frac{\sum(\text{unit_MDBF} * \text{unit_interval_count})}{\sum(\text{unit_interval_count})}$
- c. Calculate MTTR:
 - i. $\text{repair_count} = \text{number of reports in the segment}$
 - ii. $\text{total_labor_hours} = \text{sum of Labor Hours for those reports}$
 - iii. If $\text{repair_count} > 0$:
 1. $MTTR = \text{total_labor_hours} / \text{repair_count}$
- d. Calculate Availability:
 - i. If MTBF and MTTR are defined:
 1. $\text{Availability} = MTBF / (MTBF + MTTR)$

OUTPUT:

- For each segment:
 - MTBF
 - MDBF
 - MTTR
 - Availability

9.3 Reliability Trends

Pseudo code for Duane and Weibull analysis.

9.3.1 Duane

INPUT:

- problem_reports
- selected_product
- filters
- analysis_basis (which determines MTBF or MDBF)

PROCESS:

1. Fetch all problem reports associated with the selected product and all activated filters
2. Determine calculation segments:
 - a. If one or more units are selected:
 - i. create one segment per unit
 - b. Else if one or more breakdown items are selected and no units are selected:
 - i. create one segment per breakdown item
 - c. Else:
 - i. create one segment containing all reports matching the filters
3. For each segment:
 - a. Select all reports associated with that segment
 - b. Group the segment's reports by Unit ID
 - c. For each segment:
 - i. Calculate cumulative MTBF (or MDBF) for each report according to 9.2.1 MTBF, MDBF, MTTR and Availability
 - ii. Assign values to:
 1. X_i = sum of all cumulative MTBF (or MDBF) up to report i
 2. N_i = cumulative number of failures
 3. Θ_i = cumulative mean between failures:

- ii. Transform the data:
 1. $x_i = \ln(X_i)$
 2. $y_i = \ln(\Theta_i)$
- iii. Perform linear regression on:
 1. $y_i = b + \alpha * x_i$
- iv. Extract:
 1. α = slope
 2. $K = \exp(\text{intercept})$
 3. R^2 as goodness of fit
- v. Generate the Duane plot:
 1. x-axis: cumulative exposure X_i
 2. y-axis: cumulative mean between failures Θ_i
 3. plot in logarithmic scales
 4. overlay the fitted Duane model:
 - a. $\Theta = K * X^\alpha$

OUTPUT:

- For each segment:
 - Duane growth rate (α)
 - Goodness-of-fit measure (R^2)
 - Duane plot

9.2.2 Weibull**INPUT:**

- problem_reports
- selected_product
- filters
- analysis_basis (which determines MTBF or MDBF)

PROCESS:

1. Fetch all problem reports associated with the selected product and all activated filters
2. Determine calculation segments:
 - a. If one or more units are selected:
 - i. create one segment per unit
 - b. Else if one or more breakdown items are selected and no units are selected:
 - i. create one segment per breakdown item
 - c. Else:
 - i. create one segment containing all reports matching the filters
3. For each segment:
 - a. Select all reports associated with that segment
 - b. Group the segment's reports by Unit ID
 - c. For each segment:
 - i. Calculate cumulative MTBF (or MDBF) for each report according to 9.2.1 MTBF, MDBF, MTTR and Availability
 - ii. For each ordered report i , calculate the median rank:
 1. $F_i = (i - 0.3) / (n + 0.4)$
 - iii. Transform the data:
 1. $x_i = \ln(t_i)$
 2. $y_i = \ln(-\ln(1 - F_i))$
 - iv. Perform linear regression on:
 1. $y_i = b + \text{beta} * x_i$
 - v. Extract Weibull parameters:
 1. $\beta = \text{slope}$

$$2. \quad \eta = \exp(-\text{intercept} / \beta)$$

- vi. Calculate goodness of fit:
 - 1. R^2 from the linearised regression
- vii. Generate the Weibull probability plot:
 - 1. plot observed points (x_i, y_i)
 - 2. overlay the fitted straight line

OUTPUT:

- For each segment:
 - Shape parameter (β)
 - Characteristic life (η)
 - Goodness-of-fit measure (R^2)
 - Weibull plot