FEDERAL UNIVERSITY OF SANTA CATARINA
TECHNOLOGY CENTER
AUTOMATION AND SYSTEMS ENGINEERING DEPARTMENT
UNDERGRADUATE COURSE IN CONTROL AND AUTOMATION ENGINEERING

Bruno Tavares dos Anjos

Automatic detection of faults and possible optimizations in machining processes

Aachen
2024

Bruno Tavares dos Anjos

Automatic detection of faults and possible optimizations in machining processes

Final report of the subject DAS5511 (Course Final Project) as a Concluding Dissertation of the Undergraduate Course in Control and Automation Engineering of the Federal University of Santa Catarina.
Supervisor: Prof. Ricardo José Rabelo
Co-supervisor: Sven Schiller, M.Sc.

Aachen
2024

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Tavares dos Anjos, Bruno
Automatic detection of faults and possible optimizations in machining processes / Bruno Tavares dos Anjos ; orientador, Ricardo José Rabelo, coorientador, Sven Schiller, 2024.
56 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia de Controle e Automação, Florianópolis, 2024.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Optimization in machining processes. 3. Digital twin. 4. Fault detection. I. Rabelo, Ricardo José. II. Schiller, Sven. III. Universidade Federal de Santa Catarina. Graduação em Engenharia de Controle e Automação. IV. Título.

Bruno Tavares dos Anjos

Automatic detection of faults and possible optimizations in machining processes

This dissertation was evaluated in the context of the subject DAS5511 (Course Final Project) and approved in its final form by the Undergraduate Course in Control and Automation Engineering

Florianópolis, December 19th, 2024.

Prof. Marcelo De Lellis Costa de Oliveira, Dr.
Course Coordinator

Examining Board:

Prof. Ricardo José Rabelo, Dr.
Advisor
UFSC/CTC/EAS

Sven Schiller, M.Sc.
Supervisor
gemineers GmbH

Prof. Carlos Barros Montez, Dr.
Evaluator
UFSC/CTC/EAS

Prof. Hector Bessa Silveira, Dr.
Board President
UFSC/CTC/EAS

This work is dedicated to my classmates, my family,
especially my parents and my brother and my lovely
girlfriend.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to all those who supported me throughout the development of this project.

First, my sincere thanks to Victor and Icaro for their invaluable help in planning and executing the project. To Sven, my company supervisor, for his guidance and insights and to my friends at the company — Mauricio, Arthur, Pedro, Giovanna, Lucas, Milena, Fernando, Maurici, Rhanna, Maria, Piccoli, Allan, Paulo and Luiz — for their friendship and support.

To my family, especially my parents - Dilene and Alexandre - and my brother - Daniel -, for their unwavering love and encouragement and to my girlfriend - Sofia - for her patience, love and motivation throughout this journey.

Finally, I am deeply thankful to Tommy and Vincent for providing me with this incredible opportunity and experience.

DISCLAIMER

Aachen, December 19th, 2024.

As representative of the gemineers GmbH in which the present work was carried out, I declare this document to be exempt from any confidential or sensitive content regarding intellectual property, that may keep it from being published by the Federal University of Santa Catarina (UFSC) to the general public, including its online availability in the Institutional Repository of the University Library (BU). Furthermore, I attest knowledge of the obligation by the author, as a student of UFSC, to deposit this document in the said Institutional Repository, for being it a Final Program Dissertation (*“Trabalho de Conclusão de Curso”*), in accordance with the *Resolução Normativa n° 126/2019/CUn*.

Sven Schiller

Sven Schiller
gemineers GmbH

ABSTRACT

The CAM Star project was developed to address the need for advanced monitoring, fault detection and optimization in complex machining processes, driven by industrial demands for precision and efficiency. Advanced monitoring enables tracking key parameters like tool wear, temperature and vibration, ensuring that operators gain immediate insight into process conditions. Fault detection identifies deviations from expected performance, such as excessive load or unusual tool behavior, allowing swift corrective actions to prevent damage or quality issues. Optimization focuses on adjusting machining parameters to enhance efficiency, extend tool life and minimize waste while maintaining the high precision required in modern manufacturing. The system leverages real-time data analysis and process simulations to provide actionable insights and decision support. Built with a scalable and distributed architecture, it employs a modular framework for task management and data processing. The development followed an iterative and incremental methodology, ensuring continuous refinement and delivery of a user-friendly, robust application. Its architecture organizes the system into specialized components to handle data flow, processing and storage efficiently. Operators can initiate tests, view key performance indicators and monitor performance metrics in real time without requiring advanced technical expertise. The task management system enables parallel test execution, with independent task handlers ensuring efficient and scalable operations. Key outcomes of the project include reliable real-time test feedback, enhanced scalability and an intuitive interface, allowing operators to continuously monitor and optimize machining processes. While CAM Star has significantly improved process quality and operational efficiency for gemineers GmbH and its clients, the system's performance is inherently influenced by the quality of input data and the accuracy of simulation models. Further development is required to increase its adaptability and expand its applicability to diverse manufacturing contexts.

Keywords: Optimization in machining processes. Digital twin. Fault detection.

RESUMO

O projeto CAM Star foi desenvolvido para atender à necessidade de monitoramento avançado, detecção de falhas e otimização em processos de usinagem complexos, impulsionada pela demanda industrial por precisão e eficiência. O monitoramento avançado permite o acompanhamento de parâmetros-chave, como desgaste da ferramenta, temperatura e vibração, garantindo que os operadores tenham uma visão imediata das condições do processo. A detecção de falhas identifica desvios no desempenho esperado, como carga excessiva ou comportamento incomum da ferramenta, permitindo ações corretivas rápidas para evitar danos ou problemas de qualidade. A otimização, por sua vez, concentra-se em ajustar os parâmetros de usinagem para melhorar a eficiência, prolongar a vida útil das ferramentas e minimizar o desperdício, mantendo a alta precisão exigida na fabricação moderna. O sistema utiliza análise em tempo real e simulações de processos para fornecer insights acionáveis e suporte à tomada de decisões. Desenvolvido com uma arquitetura modular e escalável, ele adota uma abordagem distribuída para o gerenciamento de tarefas e processamento de dados. O desenvolvimento seguiu uma metodologia incremental e iterativa, assegurando refinamento contínuo e entrega de uma aplicação robusta e fácil de usar. Sua arquitetura organiza o sistema em componentes especializados, otimizando o fluxo de dados, processamento e armazenamento. Os operadores podem iniciar testes, visualizar indicadores-chave de desempenho (Key Performance Indicators (KPIs)) e monitorar métricas de desempenho em tempo real, sem a necessidade de conhecimento técnico avançado. O sistema de gerenciamento de tarefas permite a execução paralela de testes, com processadores independentes garantindo operações eficientes e escaláveis. Os principais resultados do projeto incluem feedback confiável em tempo real, alta escalabilidade e uma interface intuitiva que permite aos operadores monitorar e otimizar continuamente os processos de usinagem. Embora o CAM Star tenha trazido melhorias significativas na qualidade e eficiência operacional para a gemineers GmbH e seus clientes, o desempenho do sistema é influenciado pela qualidade dos dados de entrada e pela precisão dos modelos de simulação. O desenvolvimento contínuo é necessário para ampliar sua adaptabilidade e expandir sua aplicação a diferentes contextos de fabricação.

Palavras-chave: Otimização em processos de usinagem. Gêmeo digital. Detecção de falhas.

LIST OF FIGURES

LIST OF FRAMES

LIST OF ABBREVIATIONS AND ACRONYMS

API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
IPW	In-Process Workpiece
KPIs	Key Performance Indicators
MVP	Minimum Viable Product
PMED	Pseudo-Median
UI	User Interface

CONTENTS

1 INTRODUCTION

In the realm of advanced manufacturing, machining processes represent a critical element in producing high-precision parts for industries such as aerospace, automotive and energy. At the heart of these processes are constant efforts to improve efficiency, precision and predictability, all while reducing waste and downtime. This final project report addresses a significant part of this challenge by developing a solution aimed at automating and optimizing the validation of machining processes at gemineers GmbH, a company focused on advanced manufacturing software and technologies.

The gemineers GmbH company provides software solutions and services designed to enhance efficiency and quality control in the manufacturing sector. This project aligns with the company's broader goals by addressing process validation, particularly within its CAM Star system — a tool aimed at supervising and improving machining processes. In this context, the primary focus of this work is to design a solution that can streamline and automate the testing of machine process conditions, thereby reducing manual oversight and supporting the detection of performance issues in real-time. This solution is anticipated to significantly impact gemineers GmbH's ability to ensure process reliability, provide high-quality assurance for clients and support more robust data-driven insights.

1.1 PROBLEM OVERVIEW AND IMPORTANCE

In the traditional manufacturing validation workflow, testing machine processes often involves manual monitoring, relying on static thresholds and basic metrics, which can miss critical dynamic behaviors in the machining environment. These limitations increase the risk of tool degradation and process failure, leading to unplanned downtimes, costly repairs and potential risks to part quality and precision. Additionally, current validation techniques may not fully leverage data analytics, limiting the depth and accuracy of insights drawn from machine performance.

The need for a more comprehensive and automated solution to monitor machine processes in real-time is critical for gemineers GmbH and their clients. By achieving a solution that accurately and autonomously detects and alerts for deviations in machine process parameters, the project aims to reduce downtime, improve maintenance scheduling and enhance overall operational efficiency.

1.2 OBJECTIVES

The project's objective can be stratified into general and specific goals. The general objective is to develop a system capable of real-time validation and assessment of machining processes within the CAM Star environment, integrating analytics for improved machine oversight. Specific objectives include:

- Developing a back end capable of handling diverse signal analyses, implementing complex validation tests on multiple machine parameters such as chip depth, spindle speed and contact time;
- Implementing a front end interface to provide operators with real-time feedback on machine status, visualizing tests and results and enabling rapid corrective action;
- Establishing an architecture that supports modular test definitions, allowing for easy addition and customization of validation tests;
- Enabling data persistence in a way that ensures test traceability, historical analysis and future data modeling applications.

1.3 PROPOSED SOLUTION OVERVIEW

To achieve these objectives, this project presents a multi-component solution that includes:

- A **Back end system** that handles the processing, scheduling and monitoring of machine parameter validation tests. The back end is designed to process multiple streams of machine data, conduct real-time analysis against predefined thresholds and identify deviations that may indicate process anomalies.
- A **Streamlit-based front end** MVP, enabling users to initiate tests, monitor progress and visualize results directly. This interface is designed for simplicity and ease of use, particularly suitable for operators who may not require deep technical knowledge.
- A **modular test framework** where each test can independently assess conditions based on predefined thresholds, such as chip contact time or spindle load. This framework facilitates expansion by allowing additional tests to be seamlessly integrated.

1.4 METHODOLOGY AND TOOLS USED

The project employs several technical methods and tools:

- **Data Analysis**: Signal processing techniques, including Blackman window filtering and median filtering, are applied to detect and smooth noise in time-series data collected from machining processes.

- **Back end Architecture**: Implemented using an asynchronous framework for request handling, a non-relational database for flexible data management and a message broker for task queuing and real-time status updates.
- **Front end Interface**: Designed using Streamlit to provide an MVP interface for test monitoring and visualization, allowing for real-time updates on test results and machine process parameters.

1.5 RESULTS AND IMPACT

The project successfully achieved its objectives by developing a back end validation solution that autonomously monitors and assesses machine process parameters within CAM Star. This system enables more accurate and timely detection of issues, thereby supporting predictive maintenance and minimizing downtime. The implemented front end MVP allows operators to intuitively view test results, increasing responsiveness and reducing the need for manual intervention.

The impact of these results is notable, as the system provides gemineers GmbH and its clients with a proactive approach to machine process validation. By reducing downtime, extending tool life and improving part quality, the system offers significant operational and cost-saving benefits. Additionally, the approach aligns with data-driven manufacturing trends, enabling the generation of rich historical datasets that can support future insights and optimizations.

1.6 DOCUMENT STRUCTURE

The remainder of this document is organized as follows:

- **Chapter 2** describes the methodology, detailing the structured approach, data processing techniques and key tools used to build the solution.
- **Chapter 3** covers the theoretical background, including key concepts in signal processing, test validation frameworks and data management relevant to this project.
- **Chapter 4** elaborates on the specific problem and technical requirements, providing a deep dive into the challenges addressed by the system.
- **Chapter 5** presents the proposed solution, detailing the architectural design, component interactions and core functionalities within the CAM Star system.
- **Chapter 6** focuses on the implementation process, discussing both back end and front end components, along with the tests created and the results observed.

- **Chapter 7** concludes the project by summarizing the results, discussing limitations and offering suggestions for future improvements.

This structured approach provides a comprehensive view of the project, from problem context to solution deployment, underscoring the innovations and impacts made through this work.

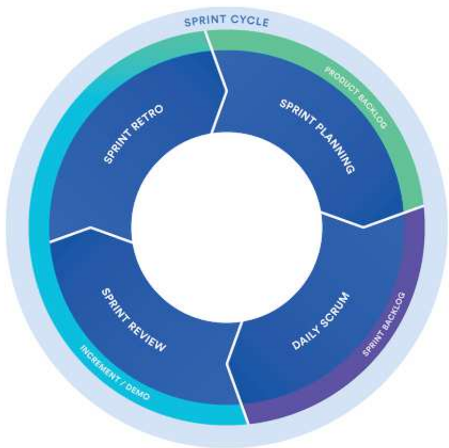
2 METHODOLOGY

A development of the CAM Star project followed the agile Scrum methodology, reflecting the collaborative culture of gemineers GmbH, where Scrum is widely used to promote flexibility, teamwork and iterative progress in software development. This chapter outlines the key elements of Scrum as applied to CAM Star and provides an overview of the iterative approach taken to build, test, and refine the system, ensuring continuous improvement. A detailed explanation of the structured project methodology was removed for confidential purposes.

2.1 SCRUM METHODOLOGY

Scrum is an agile framework that promotes incremental development through short, focused work cycles called sprints, where the team sets specific goals. The framework includes key rituals such as daily stand-ups, sprint planning, sprint reviews and retrospectives, all designed to facilitate communication and adaptability.

Figure 1 – Scrum cycle structure.



Source: Atlassian website (Atlassian, 2023)

2.1.1 Sprint Planning

At the start of each sprint, a sprint planning session was held to define the tasks that would be completed during the cycle. Each team within **gemineers GmbH** worked independently to establish their priorities, aligning their objectives with overall project goals. During these sessions, the team used planning poker to estimate the effort and complexity of each task, balancing both the technical challenges and time required to complete them. This process ensured that each sprint had realistic and achievable targets, facilitating smooth progress throughout the project.

2.1.2 Daily Stand-ups

Daily stand-ups or "dailies" are brief, time-boxed meetings (usually 15 minutes) where team members share updates on their progress, plans for the day and any obstacles they are facing. During the CAM Star project, these meetings fostered real-time communication and early identification of blockers, allowing the team to swiftly address issues and adapt to changing requirements. The constant feedback loop helped maintain momentum and enabled quick course corrections, a hallmark of agile practices (Schwaber, 2017).

2.1.3 Sprint Review

At the end of each sprint, a sprint review was conducted, during which completed tasks were demonstrated and evaluated. Stakeholders, including management at **gemineers GmbH**, participated in the reviews to provide feedback on the system's current state. This iterative feedback ensured that the CAM Star system continuously evolved to meet user needs and expectations. The feedback loop also facilitated fine-tuning of the system's features and functionalities to ensure alignment with the overall project goals (Schwaber; Sutherland, 2019).

2.1.4 Sprint Retrospective

Following the sprint review, the team engaged in a sprint retrospective to reflect on the sprint's successes and challenges. Retrospectives aimed to identify areas for improvement in team processes and workflows. These sessions were pivotal in enhancing communication and refining task prioritization strategies, ensuring that the team improved efficiency with each successive sprint. This practice of continuous reflection and adaptation helped maintain high productivity levels throughout the project (Sutherland; Schwaber, 2014).

3 THEORETICAL BACKGROUND

This chapter explores the foundational concepts to understanding the CAM Star project, including machining processes, digital twins, simulation data, signal processing, tool limits, HTTP protocols, message queuing, priority-based scheduling and parallel processing. Detailed discussions about specific technologies were removed for confidential purposes.

3.1 MACHINING PROCESSES

Machining is a fundamental manufacturing process that removes material from a workpiece to achieve desired geometries, tolerances and surface finishes. It is widely used across industries, from automotive to aerospace, to produce parts with high precision and intricate features. Common machining processes include milling, drilling and turning, each of which has specific mechanisms and applications.

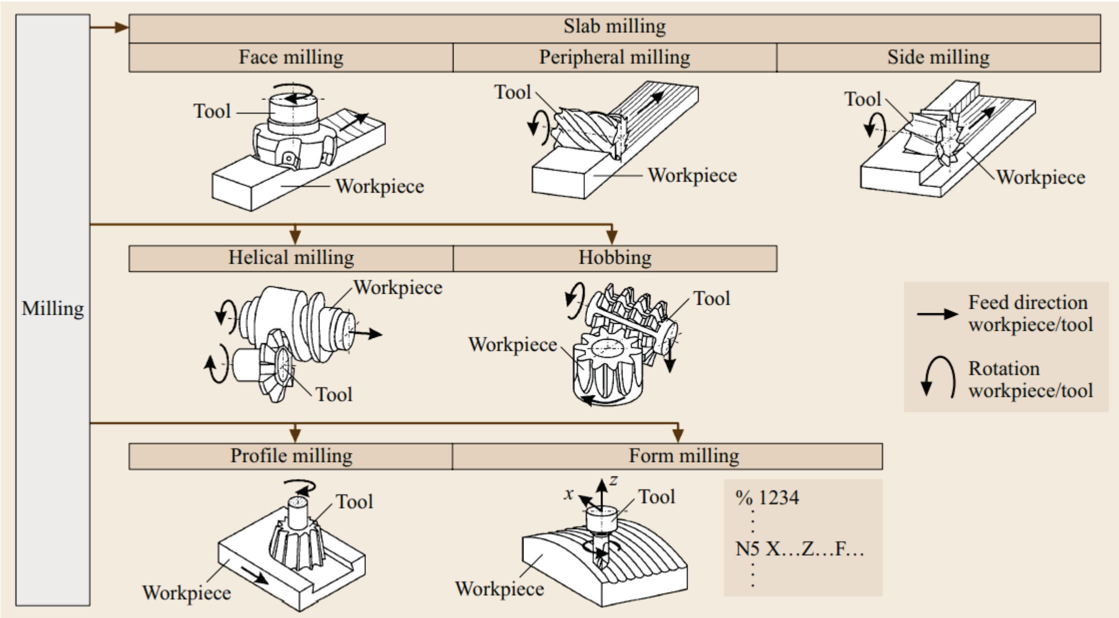
3.1.1 Milling

Milling is a versatile machining process where a rotating tool with multiple cutting edges removes material from a stationary or moving workpiece. Milling operations can produce complex shapes and high-precision features due to the flexibility of controlling cutting parameters such as spindle speed, feed rate, and depth of cut. The key types of milling include:

- **Face Milling**: The cutting tool engages the workpiece with its face, creating flat surfaces. It is commonly used for creating planar faces perpendicular to the tool axis.
- **End Milling**: In this operation, the cutting edges are on the periphery and end of the tool, allowing for the creation of slots, pockets and contours.
- **Peripheral Milling**: The tool engages the workpiece with its side, removing material along the tool's axis to produce profiles or contours.

Milling's effectiveness depends on tool geometry, material properties and cutting conditions. Choosing optimal parameters can significantly affect material removal rates, surface finish and tool wear (Kalpakjian; Schmid, 2001; Degarmo; Black, J. T.; Kohser, 2011). This versatility makes milling suitable for a broad range of applications, including aerospace and automotive component manufacturing.

Figure 2 – Representation of milling processes.



Source: (Karpuschewski *et al.*, 2021)

3.1.2 Drilling

Drilling is a specialized machining process that creates circular holes in a workpiece using a rotating drill bit. It is the most efficient method for creating cylindrical holes and is widely used for tasks that demand precision, such as installing bolts or rivets. Drilling operations are influenced by factors like drill bit geometry, material hardness and cutting parameters. The main types of drilling include:

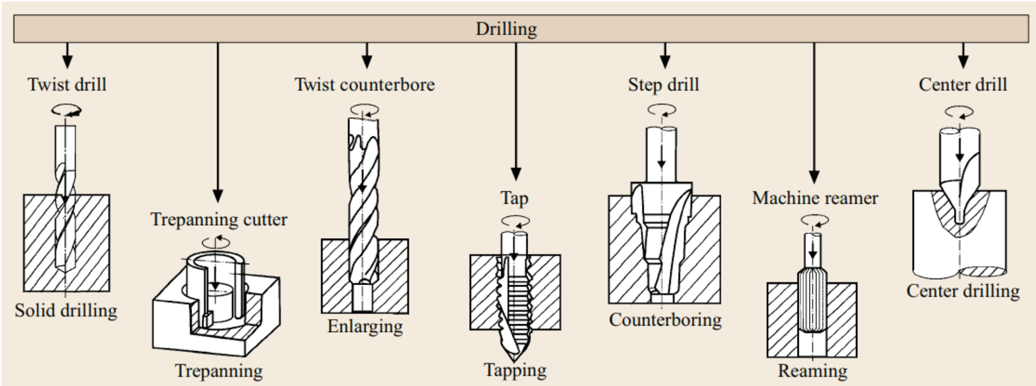
- **Spot Drilling**: A preliminary step where a short hole is drilled to guide subsequent operations, ensuring accuracy and preventing the drill from wandering.
- **Deep Hole Drilling**: Specialized for holes with a depth greater than ten times the diameter, often requiring coolant flow for chip evacuation and temperature control.
- **Countersinking and Counterboring**: Processes to create tapered or cylindrical spaces at the hole entrance to accommodate screws or fasteners.

Drilling can involve significant axial forces, leading to tool wear and heat generation. Control of feed rate and spindle speed is crucial for prolonging tool life and ensuring precision (Shigley; Mischke, 2015).

3.1.3 Turning

Turning is a machining process where the workpiece rotates against a stationary single-point cutting tool to remove material. This process is ideal for creating cylindrical

Figure 3 – Representation of drilling processes.



Source: (Karpuschewski *et al.*, 2021)

or conical shapes with tight tolerances, commonly used in applications such as shafts and bearings. The primary types of turning operations include:

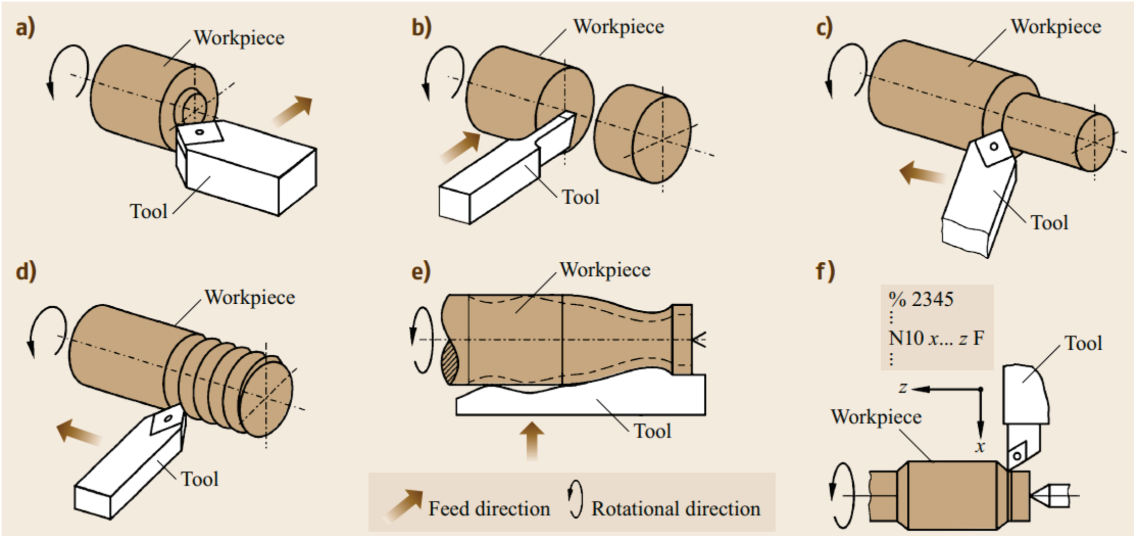
- **Straight Turning**: The tool moves parallel to the workpiece axis, producing a cylindrical surface.
- **Taper Turning**: The tool moves at an angle to the workpiece axis, creating conical surfaces.
- **Facing**: The tool moves perpendicular to the workpiece axis, creating a flat surface on the end of the workpiece.

Turning operations require precise control of factors such as cutting speed, depth of cut and feed rate to achieve the desired surface finish and dimensional accuracy. Material properties, tool wear and cooling conditions significantly influence the effectiveness of turning operations. High-speed and precision turning techniques are commonly employed in industries requiring high-quality surface finishes, such as the aerospace sector (Groover, 2016; Coromant, 2010).

3.2 DIGITAL TWINS AND SIMULATION DATA

A digital twin is a virtual replica of a physical object, process or system that mirrors real-world behavior in real time, primarily using data from sensors on its physical counterpart. In manufacturing and machining, digital twins have become essential for continuous monitoring, analysis and optimization of processes. By leveraging simulation data, digital twins can provide predictive insights and support decision-making, ultimately enhancing productivity and efficiency. Parameters like feed rate, chip contact area and temperature are dynamically analyzed through simulations to ensure optimal performance and minimize downtime (Tao *et al.*, 2018).

Figure 4 – Representation of turning processes.



Source: (Karpuschewski *et al.*, 2021)

3.2.1 Simulation Data and Signals

Simulation data plays a vital role in capturing the real-time performance of machining processes or to identify possible optimizations on it. Through signals derived from this data, operators can monitor key aspects of machining, enabling fine-tuned adjustments for enhanced precision and operational efficiency. Key signals include:

- **Feed Rate**: Feed rate, the rate at which material is fed into the cutting tool, is crucial in balancing efficiency with tool wear. By adjusting feed rates based on real-time analysis, it is possible to maintain a steady cutting speed that minimizes thermal stress and tool degradation, enhancing both tool life and machining accuracy (Davim, 2011).
- **Chip Contact Area**: The chip contact area is the interface between the cutting tool and the workpiece material. It provides critical data on the engagement region's total surface area, which affects friction, heat generation and overall tool load. Monitoring this metric can help avoid excessive wear on the tool's cutting edge and predict necessary maintenance intervals.
- **Spindle Load**: Spindle load refers to the load experienced by the spindle, often measured by the number of revolutions the milling tool makes per unit time (RPM or Hz). High spindle load may indicate excessive material resistance or suboptimal tool speed settings, potentially leading to tool breakage or compromised surface quality. By monitoring spindle load, it is possible to adjust tool speed and feed

rate to optimize machining efficiency and maintain tool health (Black, S. C. *et al.*, 1996).

- **Temperature**: Temperature is a critical variable in machining, as excessive heat can reduce tool life and affect product quality by altering material properties. Continuous monitoring of temperature signals enables real-time adjustments in coolant application, feed rate and cutting speed to prevent overheating. This ultimately extends tool lifespan, reduces the risk of defects and maintains the desired material properties in the finished workpiece (Grieves; Vickers, 2017).

3.2.2 Tool Limits: Chip Depth and Chip Width

In machining, understanding and controlling chip depth and chip width are important for managing tool loads and ensuring effective material removal rates. These parameters directly impact tool wear, heat generation and overall machining efficiency. Key considerations include:

- **Chip Depth**: Chip depth, or the penetration depth of the cutting tool, is a primary factor in determining material removal rate. Excessive chip depth can lead to increased cutting forces, tool deflection and, ultimately, breakage, particularly with brittle materials. Shallow cuts, while reducing tool load, may compromise efficiency and prolong machining time. Balancing chip depth is thus essential for maximizing material removal rates while minimizing tool wear (Tlusty, 2000).
- **Chip Width**: Chip width, the radial width of the cut, affects the amount of material engaged by the tool and, consequently, the heat generated at the cutting edge. Wider chips typically increase the thermal load on the tool, accelerating wear and potentially degrading surface finish. By adjusting chip width based on material properties and cooling capabilities, operators can control heat buildup, reduce tool wear and improve surface quality in machined parts (Boothroyd; Knight, 1989).

3.3 SIGNAL PROCESSING AND FILTERS

To analyze simulation data effectively, noise-reduction techniques such as the Blackman Window Filter and Median Filter are used.

3.3.1 Blackman Window Filter

The **Blackman Window Filter** is a widely used technique in signal processing for reducing spectral leakage during the analysis of discrete signals. Spectral leakage occurs when a signal is not periodic within the analysis window, causing energy from one frequency bin to spread into adjacent bins in the frequency spectrum. This effect can distort the frequency representation of the signal, leading to inaccurate results, especially in sensitive applications such as vibration analysis, acoustic measurements and temperature monitoring (Blackman; Tukey, 1958; Lyons, 2011).

The Blackman window function is defined mathematically as:

$$w[n] = 0.42 - 0.5\cos\left(\frac{2\pi n}{N-1}\right) + 0.08\cos\left(\frac{4\pi n}{N-1}\right), \quad 0 \leq n \leq N-1 \tag{1}$$

where N is the total number of samples in the window and n represents the sample index. The key characteristics of the Blackman window are its ability to reduce the side lobes of the frequency response, thereby minimizing leakage. The filter effectively smooths the transition of the signal to zero at the boundaries, which enhances frequency resolution while maintaining a good balance between main lobe width and side lobe level.

Research by Cavalcante (P. Cavalcante, 2023) highlighted the effectiveness of the Blackman Window Filter in comparison with other filtering techniques, such as the Hamming and Hann windows. The findings indicated that the Blackman window not only improved the overall signal fidelity but also provided superior performance in applications like digital twin generation for machining processes. By reducing artifacts that can arise from discrete Fourier transforms, the Blackman filter is instrumental in producing cleaner, more reliable frequency domain representations.

The Blackman window is particularly effective in contexts where high-frequency components of the signal need to be preserved while still controlling the amplitude of nearby spectral components. This is crucial for accurate diagnostics and monitoring in industrial applications.

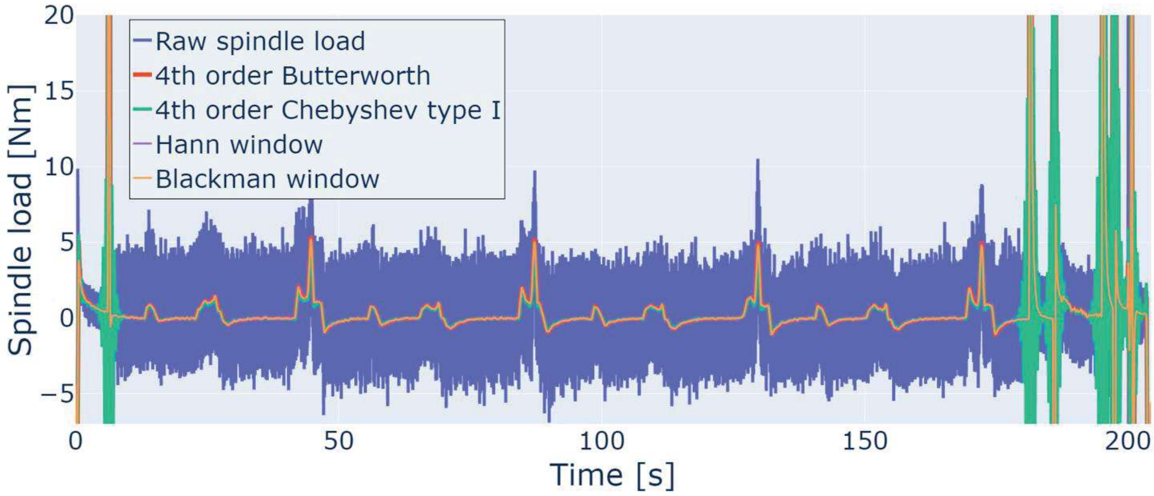
3.3.2 Median Filter

The Median Filter is a non-linear filtering technique that effectively reduces noise while preserving edges in a signal, making it particularly useful in applications such as tool wear monitoring, where sudden spikes in data can distort the accuracy of measurements (Smith, 1997; Gonzalez; Woods, 2002). The median filter works by sliding a window across the data and replacing each point with the median value within that window, reducing the impact of outliers.

In this study, a specialized version of the median filter known as the Pseudo-Median (PMED) filter is utilized. The PMED filter enhances noise reduction by calculating a value that combines the maximum of local minima and the minimum of local maxima across sliding windows, thus further preserving important signal features. The PMED calculation is as follows:

$$\text{PMED}_{S_L} = \frac{\text{MAXIMIN}_{S_L} + \text{MINIMAX}_{S_L}}{2} \tag{2}$$

Figure 5 – Comparison of filter results (Adapted from P. Cavalcante (2023)).



Source: P. Cavalcante (2023).

where:

1. **MAXIMIN** is defined as the maximum of the local minima within each sliding window of size M. 2. **MINIMAX** is the minimum of the local maxima over the same window size M.

These terms are computed as follows:

$$\text{MAXIMIN}\{S_L\} = \max\left(\{\min(S_1, \ldots, S_M), \min(S_2, \ldots, S_{M+1}), \ldots, \min(S_{L-M+1}, \ldots, S_L)\}\right) \tag{3}$$

$$\text{MINIMAX}\{S_L\} = \min\left(\{\max(S_1, \ldots, S_M), \max(S_2, \ldots, S_{M+1}), \ldots, \max(S_{L-M+1}, \ldots, S_L)\}\right) \tag{4}$$

The window size M is related to the length L of the sequence and is given by:

$$M = \frac{L+1}{2} \tag{5}$$

The PMED filter, as implemented in the ‘scipy‘ library, effectively reduces noise in the signal by averaging these calculated local extrema. This method maintains signal integrity while filtering out irregular noise spikes, which is essential in high-precision monitoring applications.

3.4 HYPERTEXT TRANSFER PROTOCOL

HTTP is the foundational protocol for data communication on the web. It supports a client-server architecture, allowing clients to request data or send information to servers. In applications like CAM Star, HTTP enables scalable and responsive systems, as multiple users can request real-time data simultaneously. HTTP is a *stateless* protocol, meaning each request-response cycle is treated independently, which contributes to its scalability (Fielding *et al.*, 1999).

3.4.1 HTTP Methods

HTTP supports several methods, each designed for specific types of operations on resources. The primary HTTP methods are:

- **GET:** The GET method is used to request data from a specific resource. It retrieves information from the server without making any modifications to the data. For example, in CAM Star, a client may use `GET` to retrieve simulation results or tool status information.
- **POST:** The POST method is used to send data to the server, often resulting in the creation of a new resource. This method can be used in CAM Star to submit test results or create new entries in the system's database.
- **PUT:** The PUT method is used to update an existing resource by replacing it with new data. It can be employed in CAM Star to update simulation parameters or configurations in the system.
- **PATCH:** Similar to PUT, the PATCH method is used to make partial updates to an existing resource, modifying only specific fields instead of replacing the entire resource. This method is suitable in CAM Star for small updates, such as changing the status of a task or updating a specific setting.
- **DELETE:** The DELETE method is used to remove a specific resource from the server. In CAM Star, DELETE might be used to delete outdated test records or remove resources that are no longer required.

3.5 WORKER IN PROGRAMMING AND CAM STAR CONTEXT

In a programming context, a **worker** is a specialized process or thread responsible for executing background tasks independently of the main application logic. Workers are crucial in distributed systems, as they handle specific, often repetitive, tasks that allow the main application to maintain responsiveness and efficiency. Typically, workers manage tasks like processing messages, performing computational operations and interacting with databases or APIs (Kleppmann, 2017; Tanenbaum; Bos, 2015).

Within the CAM Star project, a worker, as implemented in the Worker class, performs the following core functions:

- **Manages and Processes Tests:** Each Worker retrieves test instructions from a queue and executes them on various resources (e.g., machining tool data, simulation data). This system ensures that tests are run as intended on available equipment, maintaining operational flow and efficiency (Hennessy; Patterson, 2019).
- **Interacts with External Systems:** Through HTTP requests, a worker retrieves essential metadata (e.g., details on operations, processes and tests) and updates task statuses. This interaction keeps CAM Star's distributed components synchronized, ensuring real-time data availability and system consistency (Fielding *et al.*, 1999; Meszaros, 2007).
- **Reports Results and Status:** After completing each test, the worker posts the results to a central database and updates the status of tasks, which facilitates real-time monitoring, data integrity and comprehensive tracking of the system's performance (Patterson; Waterman, 2021).

The worker model, especially in distributed and high-performance computing environments like CAM Star, contributes to scalability by isolating task-specific operations from the main application thread. This design pattern enhances the system's capability to handle complex and concurrent operations, making it an effective structure for real-time and scalable systems (Sanjay Ghemawat; Leung, 2003).

4 DESCRIPTION OF THE PROBLEM AND TECHNICAL REQUIREMENTS

The CAM Star project aims to enhance machining processes by utilizing digital twins and advanced process optimization techniques. As machining operations become increasingly complex and automation continues to expand, there is a growing need for early fault detection and real-time process optimization to ensure efficient operations. Modern industrial environments face significant challenges, including reducing downtime, improving product quality and lowering operational costs, which are critical for maintaining competitiveness in the manufacturing sector (Bofill *et al.*, 2023).

A key motivation for this project is to decrease the dependence on highly experienced personnel for manual process analysis at gemineers GmbH. Currently, much of the fault diagnosis and process improvement relies on experts with deep knowledge of machining systems, creating bottlenecks when problems arise. By implementing automated fault detection systems based on digital twins, the CAM Star project seeks to democratize process management, allowing even less experienced operators to efficiently identify issues and optimize operations (Hu *et al.*, 2024).

The integration of digital twins allows the real-time monitoring and analysis of machining processes, enabling predictive maintenance and timely fault detection. This approach reduces the need for manual intervention and supports continuous improvement, ultimately leading to enhanced productivity and reduced human error.

4.1 ABOUT THE COMPANY: GEMINEERS GMBH

Gemineers GmbH - logo in Figure 6 - is a dynamic startup specialized in software solutions for the analysis and optimization of machining processes. Officially founded in July 2021, the company is based in Aachen, Germany and emerged as a spin-off from the Fraunhofer Institute for Production Technology (Fraunhofer IPT). The roots of gemineers trace back to 2019 when initial ideas for the company were developed, driven by the need for a more precise and efficient method of evaluating machining process quality within the manufacturing industry.

Figure 6 – gemineers GmbH logo.

Source: gemineers GmbH website (Gemineers, 2023)

The company's inception was fueled by extensive research carried out at Fraunhofer IPT, particularly in the field of high-precision cutting technologies. With the support of the EXIST Transfer of Research, a German government initiative that funds high-risk and innovative research projects, gemineers was able to turn its research-driven ideas into a commercially viable business. Shortly after its establishment, the startup participated in the RWTH Incubation Program, run by RWTH Aachen University, which further accelerated its growth by providing business development support.

4.1.1 Business Focus and Model

Gemineers GmbH focuses on advancing the digitization of manufacturing processes through its flagship product — an integrated Digital Twin platform for quality analysis in machining. This platform offers a comprehensive, end-to-end solution for monitoring and improving production quality, targeting manufacturers in industries that rely heavily on precision machining.

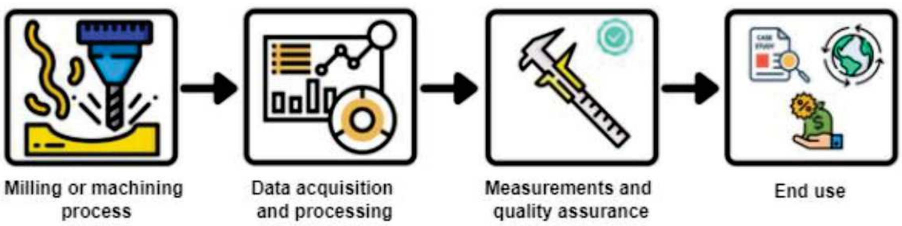
The core components of gemineers' business model include:

1. **Machine and Sensor Data Acquisition**: Providing plug-and-play data acquisition systems that are compatible with a wide range of machine controls and sensors. This allows seamless data integration from diverse production environments.

2. **Fully Automated Digital Twinning**: Creating digital twins of manufactured parts using high-performance models. These digital replicas provide insights into the real-time behavior of parts during production, enhancing the ability to predict and prevent quality issues.

3. **User-Friendly Quality Dashboard**: A browser-based interface that allows users to visualize and analyze part quality data in real time. This enables manufacturers to make faster, data-driven decisions regarding process improvements and quality control.

Figure 7 – Data acquisition and analysis workflow for the milling process.



Source: gemineers GmbH archive

4.1.2 Technological Capabilities: The Gemineers Software Platform

The gemineers software platform is a comprehensive suite for real-time data acquisition, analysis and visualization in machining operations. Its modular structure spans four core functional areas — Data Ground, Data Processing, Interfaces and Scalability Enhancements — each supporting a unique phase of the machining process chain by offering specialized data outputs and analytical functions (Mello Mattos, 2023).

4.1.2.1 Data Ground

Data Ground is the foundation of the software's workflow, focusing on acquiring raw data directly from machine tools during machining operations. This module ensures consistent and accurate capture of sensor data, including tool positions, feed rates, spindle speeds and cutting forces, across various data formats. By standardizing and formatting these inputs, Data Ground generates a unified data stream that can be processed by downstream modules. The standardized outputs from this service allow the system to maintain compatibility across different types of equipment and data types, facilitating streamlined analysis and reporting.

4.1.2.2 Data Processing

The Data Processing area refines and analyzes the standardized data for in-depth insights into machining dynamics. This area comprises several services, each handling a specific aspect of data transformation and predictive modeling.

4.1.2.3 Interfaces

The Interfaces module is responsible for presenting data to the user in a comprehensible and interactive manner, facilitating real-time decision-making. The two primary components are:

4.1.2.3.1 REST API

The REST API acts as the backbone for data integration, managing access to the database and overseeing user authentication and authorization. It organizes data flow between services and supports external system integrations.

4.1.2.3.2 Front End

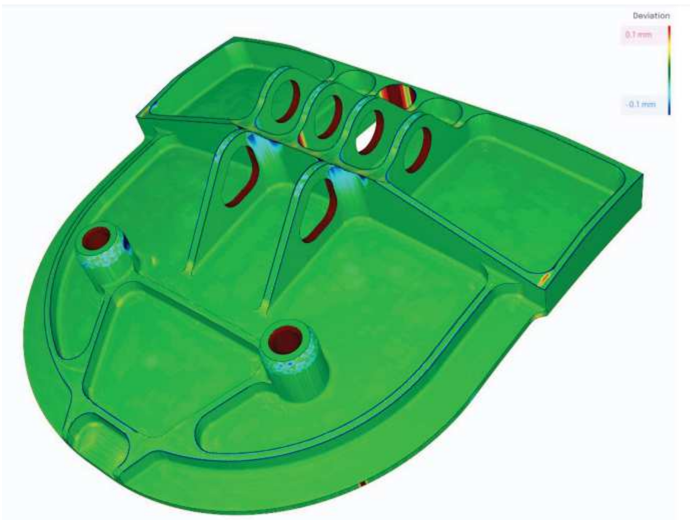
The Front End, developed with the Next.js framework, provides an intuitive, graphical interface. Key features include:

- *3D Visualization of Toolpaths and IPW Geometry*: Users can interactively view toolpath details, inspect deviations and monitor tool engagement in real time.
- *Monitoring of Machining Parameters*: Real-time feedback on metrics such as tool speed, engagement zones and deviation levels allows users to detect anomalies, like excessive tool wear or incorrect material removal, before they affect final product quality.

4.1.2.4 Scalability for Serial Production

While the gemineers platform is highly effective for individual parts, ongoing development aims to scale its capabilities to accommodate high-volume production. Enhanced data processing efficiency and batch analysis features will enable the platform to analyze quality and tool performance across numerous parts, supporting high-precision manufacturing at scale (Mello Mattos, 2023).

Figure 8 – Color-coded deviation analysis of the IPW geometry in gemineers GmbH software.



Source: (Mello Mattos, 2023)

4.2 REQUIREMENTS

This section outlines the functional and non-functional requirements of the CAM Star project. The project is divided into three main stages: CAM Star Stage 1 (4.2.1), CAM Star Stage 2 (4.2.2) and CAM Star Stage 3 (4.2.3), each with distinct goals, user stories, deliverables and out-of-scope items. This chapter will describe the requirements for each stage and the technical scope necessary to achieve the project objectives. Both stages share a common goal of enabling manufacturers to optimize machining processes and detect faults without requiring deep technical knowledge of operations, processes or tools.

4.2.1 CAM Star Stage 1 Requirements

4.2.1.1 Goal

The primary goal of Stage 1 is to create the system structure enable manufacturers to diagnose problems in their customers' products and processes efficiently. The system must be usable even by those who do not have deep knowledge of machining processes or operations. The solution should streamline test creation, execution and result analysis.

4.2.1.2 Functional Requirements

Frame 1 – Functional Requirements Description

Code	Requirement	Description
FR-1	Test Execution	The system must allow running tests for products and processes.
FR-2	Unique Tests	The system must run unique tests, where results are only the required entity.
FR-3	Cascade Tests	The system must run cascade tests, where results depend on descendants.
FR-4	Test Results Access	Users must have access to ongoing and completed test results.

Source: Personal archive.

Frame 2 – Category and Priority of Functional Requirements

Code	Category	Mandatory	Optional
FR-1	Functional	(X)	( )
FR-2	Functional	(X)	( )
FR-3	Functional	(X)	( )
FR-4	Functional	(X)	( )

Source: Personal archive.

Test Execution: The primary function of the CAM Star system is to allow manufacturers to run tests on their products and processes. This includes both single and cascade tests. Single tests are focused on one specific product or process, while cascade tests run across a resource and all its descendants. Each test generates a single result, making the output more straightforward to interpret.

Unique Tests: Unique tests are single, independent tests conducted on a specific entity, such as an operation, process or individual system component (referred to as a "gemini" within the **gemineers GmbH** context). These tests are isolated from other operations, ensuring that the results focus solely on the entity being evaluated.

Cascade Tests: Cascade tests provide deeper insights into the interdependencies between different operations. By running tests across a hierarchy of resources, the system can help identify potential issues in descendants that might affect the overall process performance. This is a crucial feature for complex manufacturing operations where multiple steps are interconnected.

Test Results Access: It is essential for users to have real-time access to the test results. During ongoing tests, users should be able to monitor the current status of tests (e.g., which ones are finished, which are in progress, which ones failed). Additionally, once a test is complete, users must have access to all the results, which can vary in format — from single values KPIs to complex charts.

4.2.1.3 Non-Functional Requirements

Frame 3 – System Non-Functional Requirements Description

Code	Requirement	Description
NFR-1	System Performance	The system must respond to user queries and test executions within 5 seconds to ensure efficient real-time decision-making during critical operations.
NFR-2	Scalability	The system should efficiently handle more then one test running in parallel.

Source: Personal archive.

Frame 4 – Category and Priority of System Requirements

Code	Category	Mandatory	Optional
NFR-1	Performance	(X)	( )
NFR-2	Scalability	(X)	( )

Source: Personal archive.

System Performance: The system must ensure that all operations, including tests and user queries, are executed in less than 5 seconds. This is critical for maintaining the efficiency of real-time operations and enabling fast decision-making, especially in environments where delays could disrupt the production flow. The system should be optimized to meet this requirement regardless the number of users or complexity of tests being run.

Scalability: Scalability is essential for supporting more than one test script for one entity without any noticeable degradation in performance. When users initiate multiple tests, the system should execute them in parallel. This parallelism improves efficiency and ensures no time is wasted waiting for one test to complete before starting another. The number of parallel executions can be easily adjusted depending on the system's load and capacity.

4.2.1.4 Out of Scope for Stage 1

Development of test scripts to be executed.

4.2.2 CAM Star Stage 2 Requirements

4.2.2.1 Goal

The primary goal of Stage 2 is to enhance the functionality developed in Stage 1 by incorporating additional features focused on optimization and integration with advanced machining processes and tools diagnostics. This stage will enable manufacturers to perform deeper analysis, including blueprint tests and provide advanced recommendations for process improvements.

4.2.2.2 Functional Requirements

Frame 5 – Functional Requirements Description

Code	Requirement	Description
FR-1	Test scripts development	Must have development of some test scripts to evaluate signals in cascade or unique mode.
FR-2	Mitigation Suggestions	The system should analyze test data and provide suggestions to optimize machining processes such as adjusting feed rates, tool paths or coolant flow.
FR-3	Process Visualization	Graphical visualization tools must be available to display test data using charts, graphs or 3D models for easier interpretation.
FR-4	Delete When Failed	The system must update related tests to "failed" if a cascade test fails and delete any succeeding successful tests to ensure consistency.

Source: Personal archive.

Frame 6 – Category and Priority of Functional Requirements

Code	Category	Mandatory	Optional
FR-1	Functional	(X)	( )
FR-2	Functional	(X)	( )
FR-3	Usability	(X)	( )
FR-4	Functional	(X)	( )

Source: Personal archive.

Test scripts: The system architecture must facilitate the development of test scripts. These tests allow developers with extensive process expertise to introduce specific testing conditions tailored to meet customer requirements or industry standards.

Mitigation Suggestions: The system should analyze test data and provide mitigation suggestions to the user. Based on test results and detected faults, CAM Star should recommend actions such as adjusting feed rates, tool paths or coolant flow to optimize the machining process based on the failed signals of the test.

Process Visualization: To enhance the interpretability of test results, Stage 2 will incorporate graphical process visualization through a suitable graphing library. This feature will present test outcomes using visual elements such as charts, graphs or even 3D models, making the analysis of blueprint tests clearer and more intuitive. By providing visual representations, the developer of the blueprint test will be able to better understand complex data, enabling easier interpretation of the test results and facilitating quicker decision-making.

Delete when failed: If a test is a standalone test (unique test) and an error occurs during the process, resulting in the test not completing, the system should update the test status to "failed." However, if the test is part of a cascade (where results depend on preceding or succeeding tests in a tree structure, the system should update all related tests in the cascade to "failed." Additionally, any tests that have succeeded within that cascade should be deleted from the database to ensure consistency (acting as a rollback mechanism).

4.2.2.3 Non-Functional Requirements

Frame 7 – Technical Non-Functional Requirements Description

Code	Requirement	Description
NFR-1	System Response Time	The system must process optimization suggestions within 3 seconds.
NFR-2	Automated Alerts	Users must receive automated alerts if any test exceeds defined thresholds or KPIs, with notifications sent via email, messaging or pop-up alerts.
NFR-3	Data Encryption	All data must be encrypted, including backups.
NFR-4	Filter Tests	The system must allow users to filter test data by entity name, ID, result status, etc.
NFR-5	Unit Testing for Mathematical Functions	All mathematical functions must have mandatory unit tests to ensure accuracy, reliability and secure usability of the system.

Source: Personal archive.

Frame 8 – Category and Priority of Technical Requirements

Code	Category	Mandatory	Optional
NFR-1	Performance	( )	(X)
NFR-2	Functional	(X)	( )
NFR-3	Security	( )	(X)
NFR-4	Usability	(X)	( )
NFR-5	Quality Assurance	(X)	( )

Source: Personal archive.

System Response Time: When providing optimization suggestions based on test data, the system must respond within 3 seconds. Faster response times allow users to make immediate adjustments to their production processes.

Automated Alerts: If any test exceeds its defined threshold — such as tool wear or a performance KPIs —, the system must generate an automatic alert. Depending on the urgency and impact level, notifications should be sent via email, a messaging system or as a pop-up notification. This ensures that operators can quickly address issues that may negatively affect production, minimizing downtime and preventing potential damage.

Data Encryption: In addition to encrypting live data transmissions, backups must also be encrypted to prevent unauthorized access to stored test results and manufacturing data.

Filter Tests: Users must be able to filter test data by various parameters such as entity name (operation, process or Gemini), ID, result status, etc., allowing for easier navigation and analysis of relevant data.

Unit Tests: Mathematical functions are critical components that must be covered by unit tests to ensure their accuracy and reliability. Unit testing of these functions is mandatory to safeguard their proper functionality and usability, ensuring that the system performs as expected under all conditions.

4.2.2.4 Out of Scope for Stage 2

- **Front end MVP:** The front end MVP must be done on stage 3. It should not be a fully developed on the current software to be deployed.

4.2.3 CAM Star Stage 3 Requirements

4.2.3.1 Goal

The primary goal of Stage 3 is to develop a fully functional front end MVP - not integrated on **gemineers GmbH** software - that enhances user interaction with the system. This stage will focus on providing an intuitive and responsive UI, enabling users to perform tasks such as running tests, visualizing data and accessing relevant KPIs, metrics and charts efficiently. All front end-related tasks and improvements will be delivered during this stage.

4.2.3.2 Functional Requirements

Frame 9 – Functional Requirements Description

Code	Requirement	Description
FR-1	Interactive Front end	The system must provide an interactive front end allowing users to initiate tests, monitor statuses and see tests results data such as charts, KPIs and test results.
FR-2	Process Visualization	The front end must include tools for graphical process visualization.
FR-3	User-Friendly Navigation	The front end must provide easy navigation between different system components.
FR-4	Dynamic Filtering	The front end must allow users to filter test data dynamically by various criteria, such as entity name, ID, result status, etc.
FR-5	Test Execution	The system must allow users to initiate and execute tests from the front end.

Source: Personal archive.

Frame 10 – Category and Priority of Functional Requirements

Code	Category	Mandatory	Optional
FR-1	Usability	(X)	( )
FR-2	Usability	(X)	( )
FR-3	Usability	(X)	( )
FR-4	Usability	(X)	( )
FR-5	Functional	(X)	( )

Source: Personal archive.

Interactive Front end: The system should provide an interactive front end that enables users to engage with various functionalities seamlessly. This includes initiating tests, monitoring the status of ongoing tests and viewing data in visual formats such as charts and KPIs. The interactive design is crucial for enhancing user engagement, making it easier for users to interact with the system intuitively and meaningfully. This allows even individuals who are not experts in machining processes to effectively analyze performance. The front end will serve as the primary UI, ensuring efficient control and real-time feedback for operations.

Process Visualization: To enhance data interpretation and decision-making, the front end should incorporate process visualization tools. This requirement includes displaying complex data in an understandable graphical format, such as 2D charts, pie charts or, for example, bar charts. These visual tools will help operators and users to monitor test data, allowing them to quickly assess the state of the processes and identify potential issues. The visualization capability also aids in spotting trends and making data-driven adjustments to optimize performance.

User-Friendly Navigation: The front end should have a straightforward and intuitive navigation system, ensuring users can easily switch between different sections of the platform, such as test results, statuses and visual data. Smooth and user-friendly navigation enhances the overall user experience by reducing the time it takes to locate key information or perform specific tasks. This requirement aims to improve productivity by ensuring that all system features are easily accessible without unnecessary complexity.

Dynamic Filtering: The system should allow users to dynamically filter test data based on various criteria such as entity name, test ID, or result status. Dynamic filtering is crucial in large datasets, enabling users to quickly find relevant information by applying specific filters. This capability will reduce time spent searching for data, increase focus on the most pertinent information and provide a more tailored view of test results and system statuses.

Test Execution: The front end should allow users to initiate and execute tests directly from the interface. During test execution, the system must provide real-time updates on test progress and performance feedback, ensuring that users remain informed at every stage. This functionality is vital as it allows users to run tests without leaving the front end, offering a seamless experience and streamlining operational workflows.

4.2.3.3 Non-Functional Requirements

Frame 11 – Technical Non-Functional Requirements Description

Code	Requirement	Description
NFR-1	System Response Time	The front end must display real-time test data and system status updates within 3 seconds.
NFR-2	Front end Performance	The front end should be lightweight, responsive and accessible on various devices with minimal loading times.
NFR-3	Scalable Architecture	The front end design should allow for scalability, enabling the system to handle increased data loads and user activity as required.

Source: Personal archive.

Frame 12 – Category and Priority of Technical Requirements

Code	Category	Mandatory	Optional
NFR-1	Performance	(X)	( )
NFR-2	Performance	( )	(X)
NFR-3	Scalability	(X)	()

Source: Personal archive.

System Response Time: The front end must respond to user interactions and display real-time test data or system status updates within 3 seconds. This is essential to ensure that users can quickly access information and make immediate decisions based on the test data. Fast response times are critical for maintaining operational efficiency, particularly in environments where timely adjustments can affect the overall output or performance of the system.

Front end Performance: The front end should be lightweight and responsive across a variety of devices, ensuring minimal loading times. Performance optimization is necessary to accommodate different users, some of whom may access the system on devices with limited computational power. A lightweight design ensures the interface can load quickly, reducing delays and improving the user experience, especially in environments with limited resources or bandwidth.

Scalable Architecture: The front end should be built with a scalable architecture, meaning it can handle increased user activity and larger data loads without a decrease in performance.

4.2.3.4 Out of Scope for Stage 3

- **Integrated Front end:** The developed front end must be only a MVP. It should not be fully developed on the current software to be deployed.

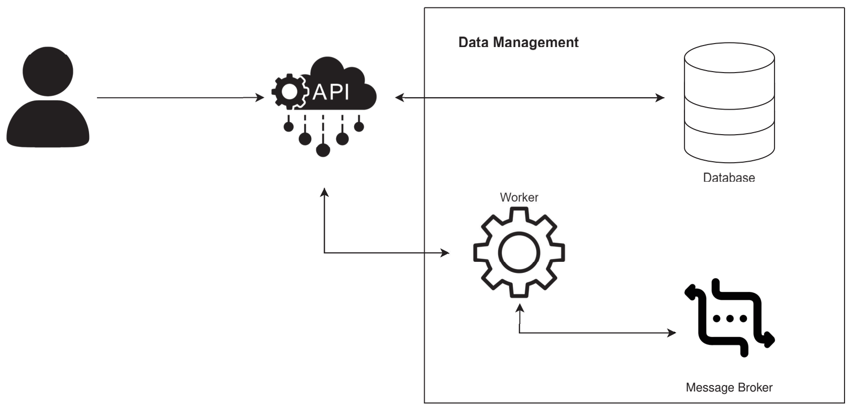
5 PROPOSED SOLUTION FOR CAM STAR

This chapter outlines the architecture of the CAM Star solution, including its key components and their interactions. While this chapter provides a high-level overview, detailed descriptions and sensitive implementation details were removed for confidential purposes. The chapter also discusses the design decisions behind the architecture, highlighting how each component meets the functional and non-functional requirements and aligns with gemineers GmbH's strategic goals.

5.1 SOLUTION ARCHITECTURE OVERVIEW

The architecture of CAM Star, shown in Figure 9, is structured as a distributed system that incorporates key components for data handling, processing and communication. The overall system flow is organized around three main layers: the UI (front end), API and Data Management layer (Daemon). This modular design ensures each component functions independently while seamlessly integrating with the overall workflow.

Figure 9 – Relationship between user and system components in CAM Star.



Source: Personal archive.

5.1.1 UI and API Layer

The user interacts with the CAM Star system through an intuitive MVP front end interface, developed as part of the Stage 3. This front end interface provides users with the ability to initiate tests, view real-time results statuses and access data visualizations, such as KPIs and test metrics, in a user-friendly format.

The API acts as the main communication hub between the user and the back end services. It enables users to:

- Send requests for test scheduling on queue.

- Retrieve test results and status updates from the database.
- Monitor ongoing tests statuses.
- Check and filter test results and statuses based on user-defined criteria.

This API layer abstracts the back end complexity, allowing users without technical expertise to operate the system efficiently. Additionally, the API's design ensures scalability, handling concurrent requests and allowing seamless interactions between components.

5.1.2 Data Management

The data management system within CAM Star is fundamental to its operation, supporting real-time monitoring, test execution and seamless communication between components. This system centers around the database and a message broker, ensuring that CAM Star can efficiently handle large volumes of data, provide responsive updates and manage task priorities. Each component is optimized for flexibility, scalability and reliability, providing robust data handling throughout the system.

6 IMPLEMENTATION AND RESULTS

The implementation was divided into three primary areas: the API, data management components (including the Worker system, message broker and the database) and the front end interface. This chapter provides a high-level overview of each component, highlighting the design principles, implementation approaches and key results.

6.1 API STRUCTURE AND ENDPOINTS

The CAM Star API is designed to provide a clear and secure interface for both user interactions and internal system processes. Comprehensive documentation ensures accessibility and clarity for end-users while safeguarding internal operations with restricted access points.

- External Endpoints: These endpoints allow end-users to initiate tests, retrieve results, monitor statuses and view KPIs and charts. External endpoints are the primary interface for user interactions, enabling seamless test management and access to critical data.
- Internal Endpoints: Restricted to system processes, these endpoints manage essential background tasks such as updating test statuses, storing intermediate results and facilitating communication between the Worker system and the database. These endpoints operate securely in the background, ensuring efficient system operations without exposing internal functionalities to users.

This structured separation ensures both robust security and system functionality, protecting sensitive operations while maintaining user accessibility to key features.

The API is further organized into two primary functional domains:

- Status API: This domain handles endpoints related to tracking and updating test statuses. It provides users with real-time insights into test progress and completion statuses, enabling comprehensive monitoring of ongoing operations.
- Test API: This segment facilitates test initiation, configuration and result retrieval. Users can start tests, set parameters and access final results, including KPIs and occurrence points, supporting detailed process analysis.

The design of these domains ensures efficient workflow management and usability.

6.2 DATA MANAGEMENT

The Data Management module in CAM Star is the backbone for handling, processing and integrating data required for test execution and result generation. It ensures seamless communication between components, efficient task queuing and resource retrieval to support diverse testing scenarios.

Key elements of the Data Management system include:

- **Worker System**: Enables parallel processing of test scripts, supporting scalability and efficient resource utilization.
- **Connection Management**: Handles communication with the task queuing system, ensuring real-time status updates, task prioritization and efficient job distribution.
- **Resource Handling**: Manages the retrieval and caching of required data, such as machining parameters, optimizing overall system performance.
- **Graph Library Integration**: Structures and organizes data, enhancing test result interpretation and supporting test script development by representing complex datasets.
- **Test Scripts**: Executes predefined logic to validate machining processes and generate actionable insights tailored to user requirements.
- **Unit Testing**: Validates the accuracy and reliability of core utility functions, ensuring the system operates as intended under all conditions.

This architecture follows a modular and distributed approach, making CAM Star highly robust, scalable and adaptable to diverse testing needs. Detailed explanations of the Worker system, data flow and other components were removed for confidential purposes.

6.2.1 Streamlit Front end MVP

The CAM Star front end MVP, developed with Streamlit, provides a streamlined interface for creating, managing and analyzing automated test results. Its design focuses on a real-time, user-friendly experience for testing, monitoring and interpreting data from manufacturing operations. The application is structured across several key pages: **Test**, **Data**, **Status**, **Test Results** and **Operations**, each serving a distinct role in the test management lifecycle.

6.2.1.1 Test Page

The Test page enables users to initiate tests for specific entity, allowing them to input parameters and configurations required for each test run. Key functionalities include:

- **Reference Selection**: Users specify the reference type and ID, such as 'operation', 'process' or 'gemini', to ensure tests are correctly assigned.
- **Test Type and User Inputs**: Options for test type (e.g., unique or cascade) and User ID to track who made the test request.
- **Signal Conditions**: If the selected test requires signal threshold conditions, users can add signals with specific comparison operators and threshold values.
- **Real-Time Monitoring**: Upon submission, the page tracks the status of each test (queued, running, success or failed) in real-time, updating users on test progress and completion.

6.2.1.2 Data Page

The Data page offers detailed access to test result data, including KPIs and visual charts. Features include:

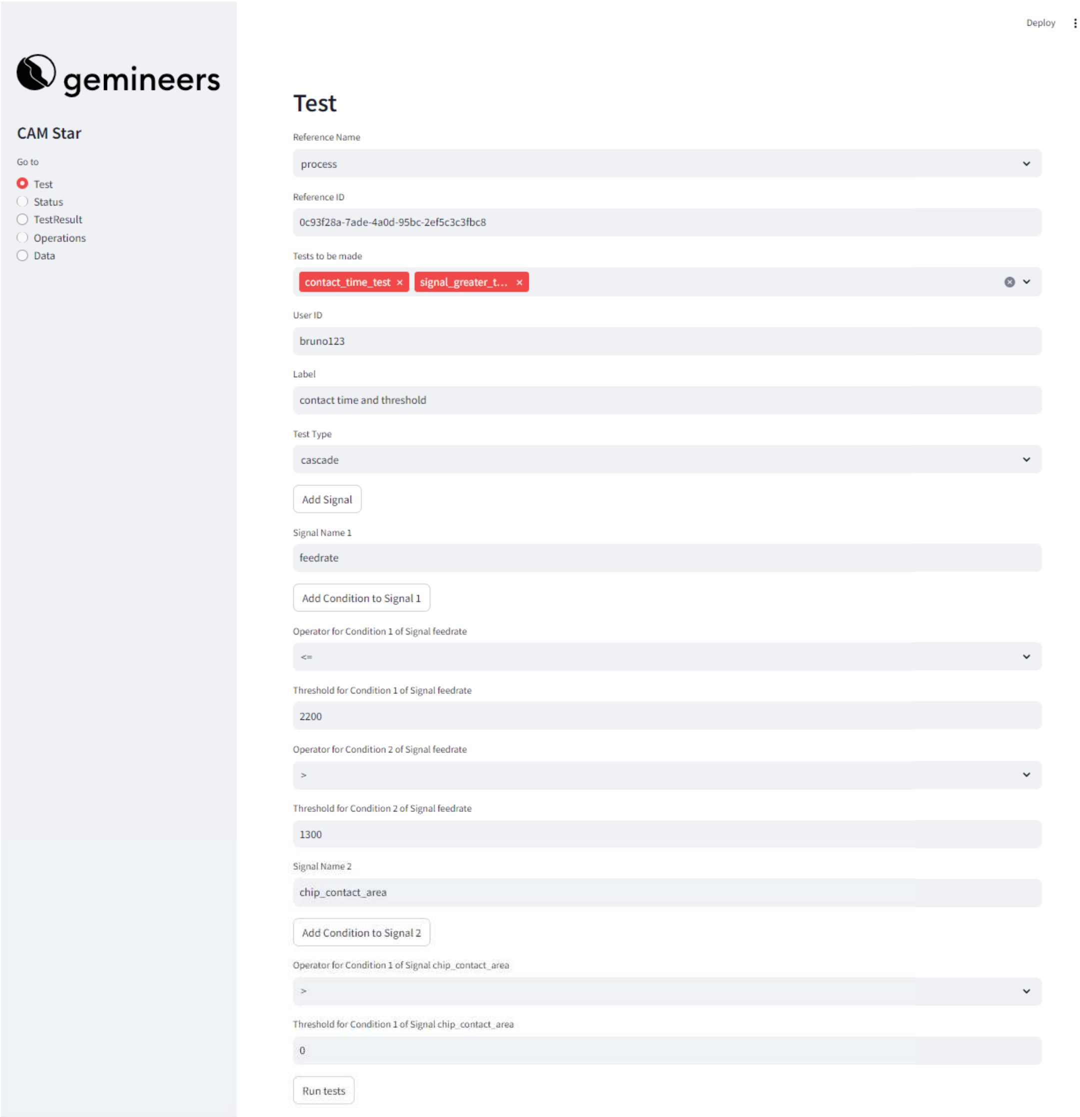
- **KPIs Display**: Presents KPIs in a structured DataFrame, allowing users to quickly analyze results.
- **Chart Visualization**: Integrates Plotly charts for various data types, including scatter, bar and pie charts. These charts provide visual insight into the test signals and highlighted failure or warning intervals.
- **Interactive Data Retrieval**: Users can choose which data artifacts (KPIs, charts, occurrence points) to display for each test, enabling customized data viewing.

6.2.1.3 Status Page

The Status page provides a summary of test statuses, allowing users to view the real-time state of multiple tests. Users can filter by:

- **Status ID, Test ID and Status**: Filters refine results by specific identifiers or status types (e.g., failed, success).
- **Real-Time Status Updates**: For queued or running tests, the page updates dynamically, showing when tests complete or fail.

Figure 10 – CAM Star MVP test page.



Source: Personal archive.

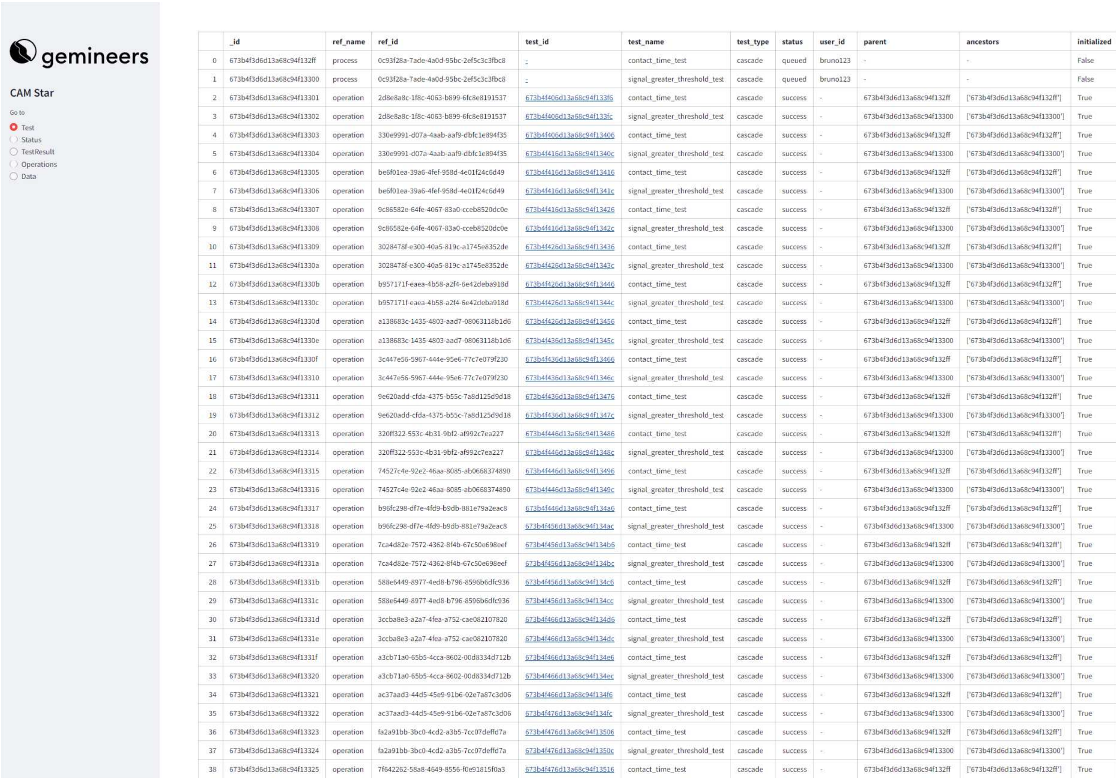
- **Data Navigation**: Interactive links direct users to the Data page for any specific test, supporting efficient navigation between status and detailed results.

6.2.1.4 Test Results Page

The Test Results page allows users to search for completed test data by various criteria:

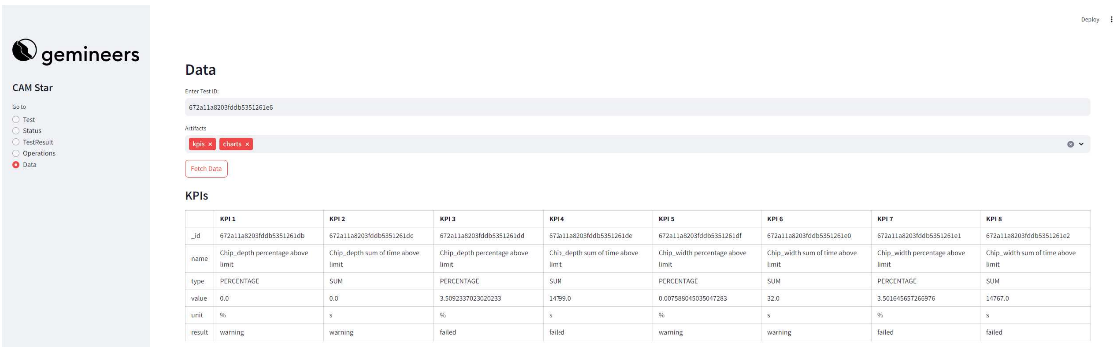
- **Reference and Test Name**: Filters by reference name or specific test names streamline results.

Figure 11 – CAM Star MVP test monitoring.



Source: Personal archive.

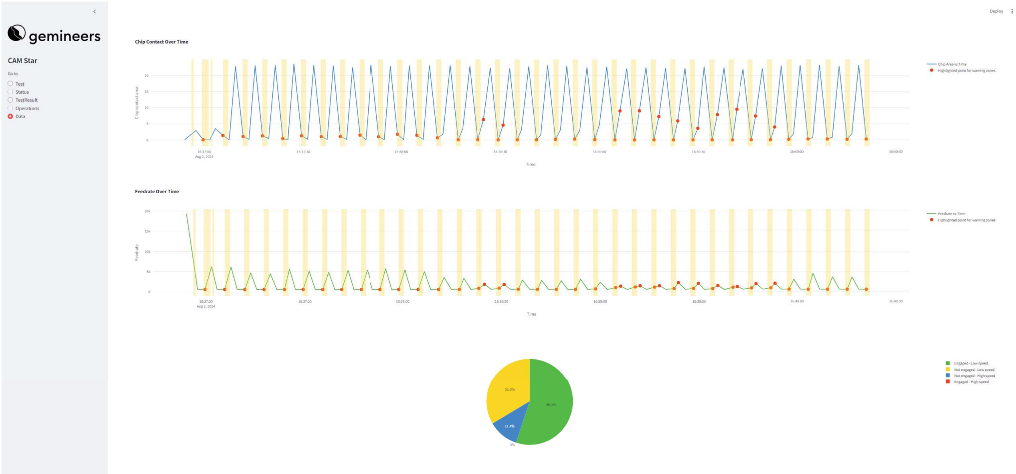
Figure 12 – CAM Star MVP Data page: KPIs.



Source: Personal archive.

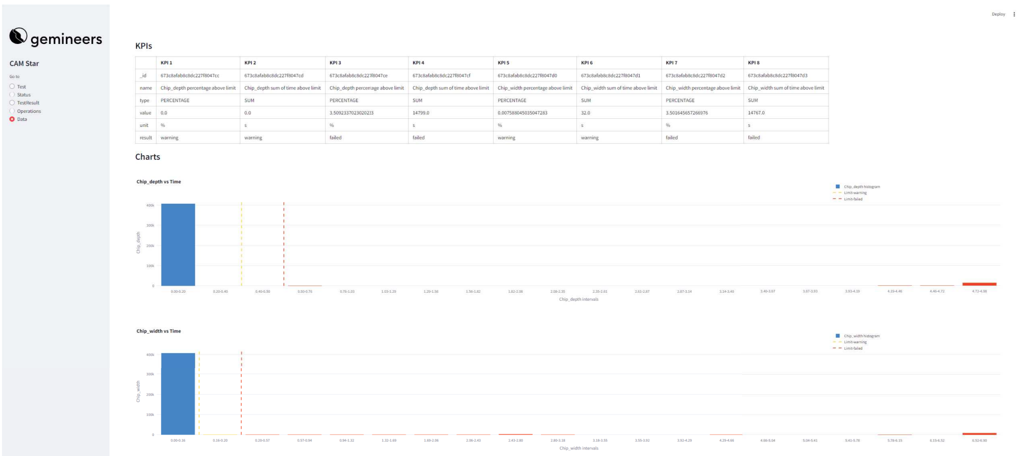
- **Mitigations, Tags, Urgency and Impact Levels**: Users can specify criteria such as mitigations tags, urgency and impact levels, enabling targeted data retrieval.
- **Detailed Display**: Search results are displayed in a DataFrame format, with interactive links to view further data details.

Figure 13 – CAM Star MVP Data page: Scatter and Pie chart.



Source: Personal archive.

Figure 14 – CAM Star MVP Data page: Bar chart.



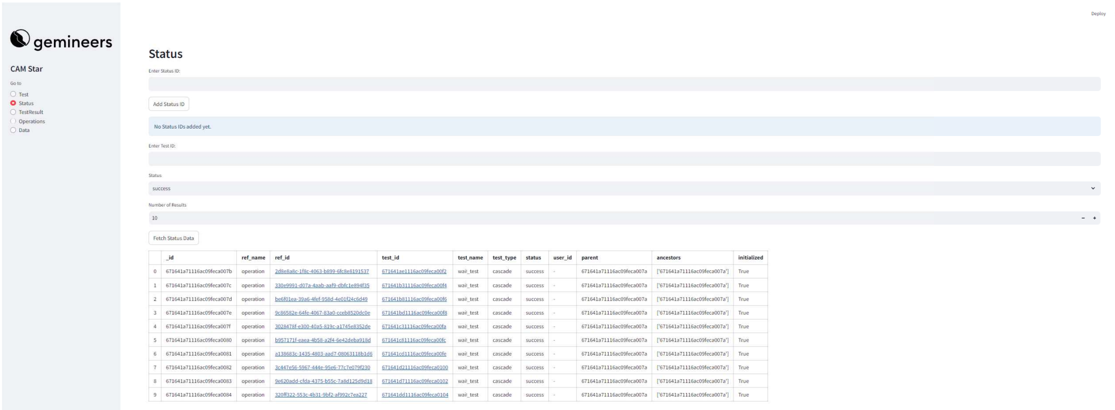
Source: Personal archive.

6.2.1.5 Operations Page

The Operations page provides access to operational metadata, facilitating a deeper understanding of each operation:

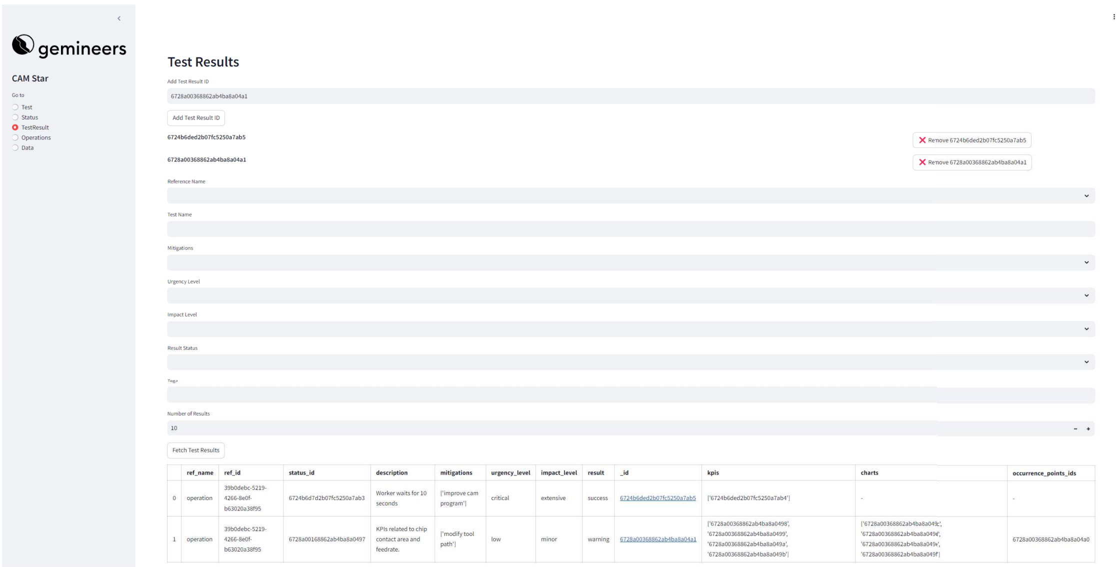
- **Operation ID Search**: Users enter an Operation ID to fetch metadata or pre-fill through query parameters.
- **Display of Operational Metadata**: Data includes key operational details like associated tools, start/end times and statuses.
- **Navigation Links**: Each Operation ID links directly to the Test page, providing seamless navigation to related test setup.

Figure 15 – CAM Star MVP status page.



Source: Personal archive.

Figure 16 – CAM Star MVP test results summary page.



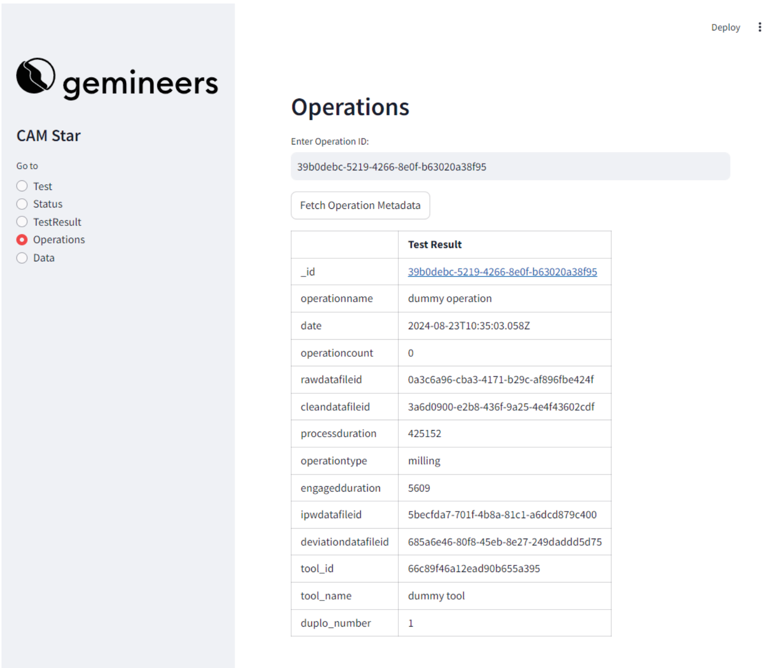
Source: Personal archive.

6.3 SUMMARY OF REQUIREMENT COMPLETION ANALYSIS

In summary, all functional requirements across Stages 1, 2 and 3 were successfully implemented and met the defined specifications. Regarding non-functional requirements:

- Stage 1: All non-functional requirements were met, ensuring system performance and scalability.
- Stage 2: Most non-functional requirements were achieved, particularly those focused on response time, filtering and unit testing.

Figure 17 – CAM Star MVP operations view.



Source: Personal archive.

- Stage 3: All non-functional requirements were fulfilled, with a scalable, responsive front end delivered as an MVP.

The completion of these requirements highlights the success of the project in meeting its goals, providing a robust foundation for further development and deployment.

7 CONCLUSION

This project addressed the challenge of automating and enhancing the test management system within CAM Star, focusing on creating a robust back end infrastructure and a functional front end interface to optimize testing workflows, improve data accessibility and ensure real-time monitoring and insights. The proposed solution was a comprehensive system that leveraged a modular architecture and user-friendly design to effectively meet these goals. The main results include a fully functional back end service with asynchronous processing capabilities, real-time status updates and a UI allowing detailed configuration, execution and monitoring of tests. Additionally, the integration of data visualization tools for KPIs and test results contributed significantly to the clarity and usability of the system.

The primary objectives of the project were achieved. Importantly, feedback from stakeholders within the company, including engineers and project managers, validated the system's success in meeting its main goals. They highlighted the practical benefits of real-time feedback, ease of use and streamlined test configurations. However, they also recognized the system as a first step — a MVP — that lays the groundwork for further development and integration into the company's main software. A key strength of the system is its potential to make testing accessible for users with minimal knowledge of machining processes, as the intuitive interface allows them to easily understand test outcomes and identify areas for improvement. This accessibility aligns with the project's goal of making advanced testing capabilities usable by a broader range of operators.

Nevertheless, certain limitations were encountered, particularly regarding the complexity of handling real-time updates and ensuring optimal synchronization across the system's components. While these challenges were addressed to a significant degree, they highlight areas that could benefit from future refinement and optimization. Moreover, the current front-end MVP provides essential functionalities but could be expanded further to enhance user navigation and support more sophisticated data visualizations.

Several improvements could be made to the project. First, transitioning the front end by integrating the service into gemineers GmbH's main product and using a framework with richer UI capabilities would allow for a more cohesive user experience, better serving users who require advanced interactions with test results and configurations. Such an integration would unify the user experience, making it easier for operators to navigate and access testing tools within a familiar platform.

In addition, optimizing data handling within the back end could enhance performance, especially for large-scale data processing during peak testing cycles. This could involve refining data structures, implementing efficient caching mechanisms or scaling database operations to handle high-throughput demands. Moreover, expanding asynchronous job handling and error reporting capabilities would provide a smoother user experience, allowing the system to proactively identify and report issues to ensure continuity in the testing process.

An advanced feature that could further enhance the project's utility is a 'test pipeline'. This would enable the chaining of tests, allowing a test to require or utilize the results from a preceding test. For instance, one test could validate basic signal thresholds, feeding its results into a subsequent, more detailed analysis on signal stability. Such pipelines would increase the flexibility and sophistication of the testing workflows, allowing users to construct layered, conditional tests that build upon each other to address complex testing scenarios.

The impacts of this project are broad and highly beneficial for the company and its stakeholders. For the organization, the automated testing system reduces manual effort, minimizes errors and ensures better adherence to testing protocols. Technologically, it provides a modular and scalable solution that is adaptable to new testing needs and future expansions. Financially, the system lowers operational costs by reducing manual workload and improving the efficiency of testing cycles. Environmentally, it supports sustainable operations by reducing rework and resource waste through early fault detection. Ethically, the enhanced quality assurance provided by this system helps ensure safety, quality and reliability for the end users and clients, ultimately benefiting the company's reputation and market position.

This project opens up several promising avenues for future work. Expanding the system's applicability to other processes within the organization, such as predictive maintenance or quality monitoring, could yield further benefits. Additionally, integrating machine learning models to analyze test results and predict potential failures or maintenance needs could enhance the system's effectiveness and reliability. Future development could also explore enhancing the visualization tools and reporting mechanisms to make data interpretation easier and more actionable for end users.

In conclusion, this project has successfully met its objectives, delivering a solution that significantly improves the CAM Star testing process and aligns with the company's operational goals. Feedback from stakeholders confirms its effectiveness as a first step toward a more comprehensive solution. The groundwork laid in this project provides a strong foundation for ongoing development and future enhancements that could expand the system's functionality and impact across the organization. By making advanced testing more accessible and intuitive, this system contributes to a broader vision of improving operational efficiency and quality assurance at gemineers GmbH.

REFERENCES

ATLASSIAN. **What is Scrum? A Guide to Scrum Methodology**. [*S. l.: s. n.*], 2023. Accessed: 2023-09-12. Available from: `https://www.atlassian.com/agile/scrum`.

BLACK, Stewart C.; CHILES, Vic; LISSAMAN, A.J.; MARTIN, S.J. 11 - Turning and Milling. *In:* BLACK, Stewart C.; CHILES, Vic; LISSAMAN, A.J.; MARTIN, S.J. (eds.). **Principles of Engineering Manufacture (Third Edition)**. Third Edition. Oxford: Butterworth-Heinemann, 1996. p. 316–371. ISBN 978-0-340-63195-9. DOI: `https://doi.org/10.1016/B978-034063195-9/50042-0`. Available from: `https://www.sciencedirect.com/science/article/pii/B9780340631959500420`.

BLACKMAN, R. B.; TUKEY, J. W. **The Measurement of Power Spectra from the Point of View of Communications Engineering**. [*S. l.*]: Dover Publications, 1958.

BOFILL, Jherson; ABISADO, Mideth; VILLAVERDE, Jocelyn; SAMPEDRO, Gabriel Avelino. Exploring Digital Twin-Based Fault Monitoring: Challenges and Opportunities. **Sensors**, v. 23, n. 16, p. 7087, 2023. DOI: `https://doi.org/10.3390/s23167087`. Available from: `https://doi.org/10.3390/s23167087`.

BOOTHROYD, G.; KNIGHT, W. A. **Fundamentals of Machining and Machine Tools**. [*S. l.*]: Marcel Dekker, 1989.

COROMANT, Sandvik. **Modern Metal Cutting: A Practical Handbook**. [*S. l.*]: Sandvik Coromant, 2010.

DAVIM, J. P. **Machining of Hard Materials**. [*S. l.*]: Springer, 2011.

DEGARMO, E. P.; BLACK, J. T.; KOHSER, R. A. **Materials and Processes in Manufacturing**. [*S. l.*]: Wiley, 2011.

FIELDING, R. T.; GETTYS, J.; MOGUL, J.; NIELSEN, H. F.; MASINTER, L.; LEACH, P. J.; BERNERS-LEE, T. Hypertext Transfer Protocol – HTTP/1.1. **Internet Engineering Task Force**, 1999. Accessed: 2023-09-12. Available from: `https://tools.ietf.org/html/rfc2616`.

GEMINEERS. **gemineers**. S.l., Oct. 2023. Accessed: 2023-10-20. Available from: `https://en.gemineers.com/`.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. [*S. l.*]: Prentice Hall, 2002.

GRIEVES, M.; VICKERS, J. **Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems**. [*S. l.*]: Springer, 2017.

GROOVER, M. P. **Fundamentals of Modern Manufacturing: Materials, Processesand Systems**. [*S. l.*]: Wiley, 2016.

HENNESSY, John L.; PATTERSON, David A. **Computer Architecture: A Quantitative Approach**: 6th ed. San Francisco, CA, June 2019.

HU, S.; LI, C.; LI, B.; YANG, M.; WANG, X.; GAO, T., *et al.* Digital Twins Enabling Intelligent Manufacturing: From Methodology to Application. **Intelligent and Sustainable Manufacturing**, v. 1, p. 10007, 2024. DOI: `https://doi.org/10.35534/ism.2024.10007`. Available from: `https://doi.org/10.35534/ism.2024.10007`.

KALPAKJIAN, S.; SCHMID, S. R. **Manufacturing Processes for Engineering Materials**. [*S. l.*]: Prentice Hall, 2001.

KARPUSCHEWSKI, Bernhard; BYRNE, Gerry; DENKENA, Berend; OLIVEIRA, João; VERESCHAKA, Anatoly. Machining Processes. *In:* **Springer Handbook of Mechanical Engineering**. Ed. by Karl-Heinrich Grote and Hamid Hefazi. Cham: Springer International Publishing, 2021. p. 409–460. ISBN 978-3-030-47035-7. DOI: `10.1007/978-3-030-47035-7_12`. Available from: `https://doi.org/10.1007/978-3-030-47035-7_12`.

KLEPPMANN, Martin. **Designing Data-Intensive Applications**. Sebastopol, CA, May 2017.

LYONS, R. G. **Understanding Digital Signal Processing**. [*S. l.*]: Pearson Education, 2011.

MELLO MATTOS, Fernando Bandeira de. **Development of an Application on a Digital Twin Platform to Optimize Cutting Tool Usage in CNC Machining**. [*S. l.: s. n.*], 2023. Accessed: 2023-09-12. Available from: `https://repositorio.ufsc.br/handle/123456789/253523`.

MESZAROS, Gerard. **xUnit Test Patterns: Refactoring Test Code**. Upper Saddle River, NJ, Apr. 2007.

P. CAVALCANTE, I. B. J. B. de. **Prediction of Cutting Forces in Milling Processes Using Acquired Machine Tool Data for Generation of Digital Twins**. [*S. l.: s. n.*], 2023. Accessed: 2023-09-12. Available from: `https://repositorio.ufsc.br/handle/123456789/248794`.

PATTERSON, David A.; WATERMAN, Andrew. **Computer Organization and Design RISC-V Edition: The Hardware Software Interface**: 2nd ed. San Francisco, CA, Jan. 2021.

SANJAY GHEMAWAT, Howard Gobioff; LEUNG, Shun-Tak. **The Google File System**. New York, NY, Oct. 2003.

SCHWABER, K. Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game. **Scrum.org**, 2017. Accessed: 2023-09-12. Available from: `https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf`.

SCHWABER, K.; SUTHERLAND, J. The Scrum Guide. **Scrum.org**, 2019. Accessed: 2023-09-12. Available from: `https://scrumguides.org/docs/scrumguide/v2019/2019-Scrum-Guide-US.pdf`.

SHIGLEY, Joseph E.; MISCHKE, Charles R. **Shigley's Mechanical Engineering Design**. [*S. l.*]: McGraw-Hill Education, 2015.

SMITH, S. W. **The Scientist and Engineer's Guide to Digital Signal Processing**. [*S. l.*]: California Technical Publishing, 1997.

SUTHERLAND, J.; SCHWABER, K. **Scrum: The Art of Doing Twice the Work in Half the Time**. [*S. l.*]: Crown Business, 2014.

TANENBAUM, Andrew S.; BOS, Herbert. **Modern Operating Systems**. Boston, MA, Aug. 2015.

TAO, F.; CHENG, J.; QI, Q.; ZHANG, M.; ZHANG, H.; SUI, F. Digital Twin-Driven Product Design, Manufacturing and Service with Big Data. **International Journal of Advanced Manufacturing Technology**, v. 94, p. 3563–3576, 2018. DOI: `10.1007/s00170-017-0233-1`.

TLUSTY, J. **Manufacturing Processes and Equipment**. [*S. l.*]: Prentice Hall, 2000.