

# **DESIGN & DEVELOPMENT OF A REAL-TIME COIL QUALITY MONITORING SYSTEM**

**Joe Andrew Frohwein**

**2315545**

**Computing (Software Engineering)**

**Tata Steel**

**Industrial Research Problem Presentation**

# Industrial Context

From steel production → inspection → operator decision

## Steel coil production

Coils must meet required mechanical and surface specifications.

## Inspection systems

IBA, Parsytec and other systems generate production and quality data.

## TLR

Current application records and presents inspection information after processing.

## Operator decision

Operator reviews the coil and determines its disposition / next route.

**Inspection data is already being generated throughout the production process, but operators currently receive the processed information after the coil has finished. This limits their ability to respond to defects while the coil is still running.**

# The Research Problem

The core problem: TLR presents inspection information only after coil processing has finished.

## **Defect detected**

A defect is identified and classified.

## **Coil processing continues**

A record of the defect is stored, and the coil continues to run through the production line.

## **Coil processing is completed**

The coil is worked in its entirety, and the entire set of coil information is reported only after processing is complete

## **Delayed operator decision**

The delayed response could potentially result in coils having to be re-run, leading to unnecessary production time and cost.

**For coils thousands of metres long, a defect discovered at the end may require the coil to be re-run so the defect can be removed and the remaining strip re-welded.**

**The problem is not necessarily defect detection, but is the delayed display of actionable information**

# Aim & Objectives

## Aim

Design and develop a real-time coil quality monitoring system that enables operators to monitor and action coils while they are being processed, reducing production delays and improving operational efficiency.

## Objectives

- Investigate the current coil-tracking system and identify limitations
- Review real-time manufacturing monitoring techniques and technologies
- Configure ROC communication and data-processing components for XML data
- Develop C# logic to validate, process and transform incoming production data
- Design and implement a real-time operator interface
- Evaluate the proposed system against the existing process

## MVP

- Receive XML datasets
- Process XML using C#
- Display live coil + defects
- Record operator actions
- Complete one production cycle

# Literature Review

## What the literature establishes

### Industrial software / SCADA

SCADA architectures demonstrate how production data from multiple sources can be acquired and presented to operators in industrial environments.

### Real-time systems

Real-time processing reduces the delay between production events, analysis and operator action. Event-driven processing suits defect and coil-length events.

### XML / data exchange

XML provides a structured, platform-independent format for exchanging data between heterogeneous manufacturing systems.

### Human-computer interaction

Operators need clear status, visible alerts and recognizable information so they can understand system state and act efficiently.

### Automated surface inspection

Vision-based systems can identify and classify steel surface defects, but large data volumes make speed, accuracy and reliability important.

**The project combines these established concepts into a real-time operator-facing monitoring solution.**

# Literature Review → Research Problem

The literature identifies a gap between defect detection and operator action

## Established by literature

Real-time manufacturing data can be processed as events occur.

Automated inspection can generate defect information.

Interfaces can present system status and alerts to support users

XML can connect heterogeneous systems.

## The gap

Detecting a defect is not enough if actionable information reaches the operator only after processing.

## Project response

Integrate existing inspection sources with the ROC framework.

Process incoming XML in real time using C# logic.

Present live coil information while the coil is still running.

Allow operator actions to be recorded

**Research problem: How can existing production and inspection data be transformed into timely, actionable information for operators?**

# Research Methodology

**Research Methodology: The systematic approach, methods and techniques used to investigate the problem**

## **Philosophy**

### **Positivism**

Measurable performance and reliability.

## **Approach**

### **Deductive**

Existing theory guides development and testing.

## **Strategy**

### **Software Development & Case Study**

Build a solution. Analyse the existing process.

## **Method**

### **Mixed Methods**

Quantitative system data & qualitative operator feedback.

## **Time Horizon**

### **Cross-sectional**

Compare current & proposed solutions within a defined period.

**To determine whether the proposed system improves processing speed, reliability, defect identification & visibility and operator usability**

# Requirements, Materials & Resources

## The requirements of the proposed system

### Must Have

- Receive XML data
- Process incoming data
- Display defect information
- Allow the operator to action coils

### Should Have

- Critical defect alerts
- Coil map

### Could Have

- Historical action log
- Advanced analytics
- Predictive quality monitoring

## The materials and resources required

### Software

- ROC framework
- Visual Studio
- C# / SQL / XML
- SQL databases

### Hardware / infrastructure

- Development PC
- Production servers
- Network infrastructure
- Inspection-system access

### People

- Software engineering support
- Production operators
- Management / stakeholders

**Requirements were identified through analysis of the existing process and consultation with operators, software engineers and management, while the solution is designed around existing Tata Steel infrastructure.**

# Data Collection & Analysis

**Quantitative: Collected through system logs and production data**

## **XML Processing time**

Mean & Comparison

## **System Availability**

Percentage Availability

## **Reliability**

Message-loss rate

## **Defect Identification**

Successful defect detection / detection performance

## **Current vs Proposed system**

Performance comparison

**Qualitative: Collected through observations and operator feedback**

## **Observations**

Used to evaluate errors, bugs and the requirements of the user

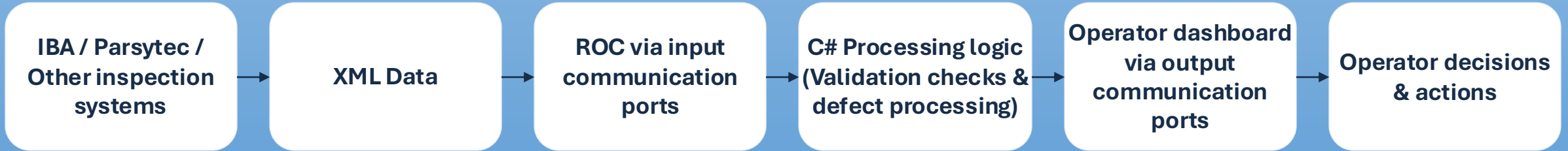
## **Operator feedback**

Used to evaluate system usability, clarity, usefulness and operator decision-making

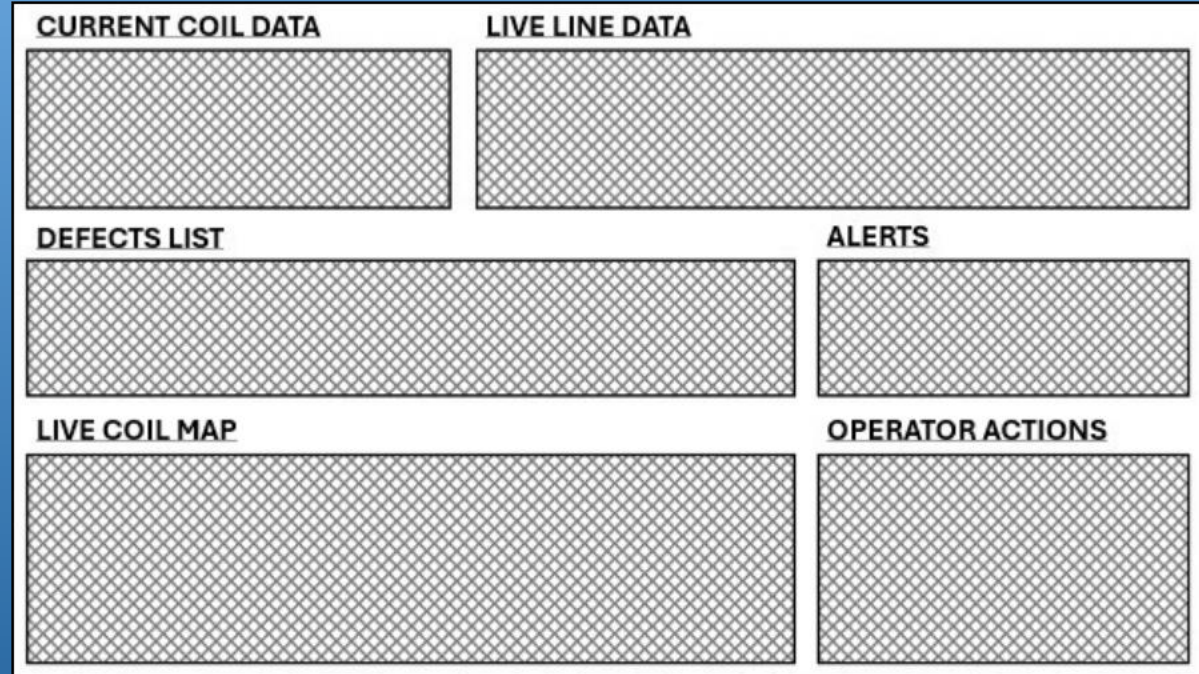
**The collected data will be used to compare the current and proposed systems against the project success criteria.**

# Project Design

## The proposed system architecture

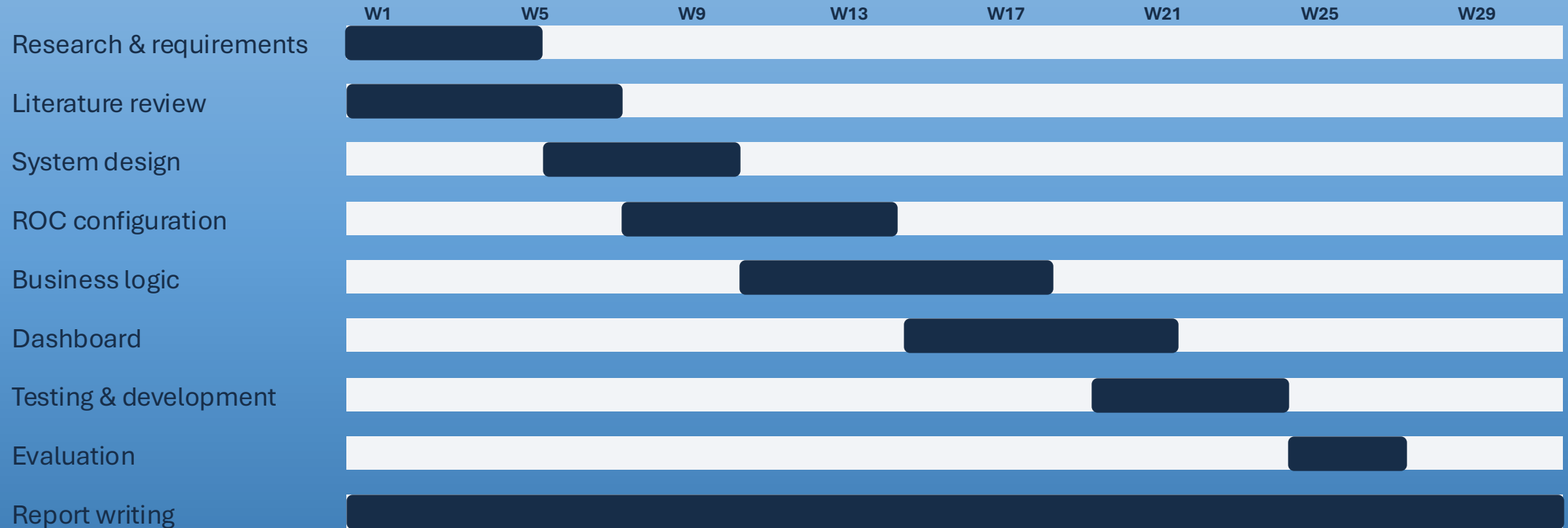


## The Operator Dashboard



# Project Planning & Funding

Planned delivery: 25 January → 29 August 2027



## Funding / capital spend

No significant additional hardware or software expenditure expected.

## Primary project cost

Development time, engineering/operator involvement, network and testing resources.

## Resource strategy

Reuse existing ROC, servers, IDEs, databases and infrastructure.

# Project Impact

## Industrial impact

- Earlier defect identification
- Faster operator decision-making
- Potential reduction in unnecessary rework / re-runs
- Improved production throughput and resource utilisation

## Limitations

- Initial deployment is limited to one production line
- Dependent on data quality and availability
- Network or recorder failures may affect live information
- Effectiveness depends on reliable integration with existing systems

## Future development

- Scale to additional production lines
- Add advanced analytics
- Investigate AI / machine learning for predictive quality monitoring

## Conclusion

**The project uses existing industrial data and infrastructure to move quality information from post-process reporting toward real-time operator action.**

# References

- [6] Yadav & Paul – Architecture and Security of SCADA Systems (2020)
- [7] Eltwab & Abdelsalam – Advanced Supervisory and Distributed Control (2026)
- [8] Sama et al. – Acquiring Automation and Control Data in The Manufacturing Industry: A Systematic Review (2023)
- [9] Hakam et al. – Advanced Monitoring of Manufacturing Process through Video Analytics (2024)
- [10] Ahmed et al. – Smart Manufacturing: MLOps-Enabled Event-Driven Architecture (2025)
- [11] Chinnappen-Rimer & Hancke – XML Model for Heterogeneous Client-Server Applications (2008)
- [12] Lau et al. – Moving XML to a Manufacturing Enterprise (2001)
- [13] Hornbaek et al. – Introduction to Human-Computer Interaction (2025)
- [14] Nielsen – 10 Usability Heuristics for User Interface Design (1994)
- [15] Neogi et al. – Review of vision-based steel surface inspection systems (2014)
- [16] Tang et al. – Review of surface defect detection of steel products based on machine vision (2023)
  
- [21] Gamage – Research Design, Philosophy, and Quantitative Approaches in Scientific Research Methodology (2025)
- [24] Ahangar et al. – Explainable AI-Driven Quality and COndition Monitoring in Smart Manufacturing (2026)
- [28] Peixoto et al. – Data Quality Assessment in Smart Manufacturing (2025)