

Digital Twin Predictive Maintenance for Manufacturing SMEsCika Amanda*, Anisa Yulianti²,¹ Informatika, Fakultas Teknik, Telkom University, Indonesia² Informatika, Fakultas Teknik, Telkom University, Indonesia

cika_cantiqu@gmail.com, yulianti_anisa808@gmail.com

Received	08 01 2026	Revised	25 01 2026
Accepted	30 01 2026	Published	02 02 2026

Abstract Digital twins can connect equipment condition, operational context, models, and maintenance decisions, but many manufacturing small and medium enterprises lack the data, integration capacity, and specialist staff required by high complexity implementations. This technical review develops a minimum viable digital twin framework for predictive maintenance. It distinguishes a digital model, digital shadow, and bidirectionally connected twin, then identifies the capabilities that produce maintenance value: asset identity, synchronized condition data, contextual history, state estimation, anomaly detection, diagnosis, prognosis, decision support, and feedback from completed work. The proposed architecture follows condition monitoring standards and separates acquisition, data manipulation, state detection, health assessment, prognostic assessment, advisory generation, and human interaction. A readiness matrix helps organizations choose between threshold monitoring, data driven models, physics based models, and hybrid approaches. The implementation roadmap begins with one consequential failure mode and a measurable maintenance decision rather than a broad platform purchase. It uses staged gates for data quality, baseline detection, shadow recommendations, workflow integration, and scaling. Unlike prior digital twin reviews that catalogue enabling technologies and definitions across manufacturing broadly, this review's contribution is a minimum-viable capability sequence and cost-aware readiness matrix scoped specifically to the resource constraints of manufacturing SMEs, where a full high-fidelity twin implementation is often not viable. The review emphasizes lifecycle governance, model uncertainty, configuration management, and the cost of maintaining the twin itself. The main conclusion is that a useful manufacturing digital twin is not defined by visual realism or data volume. It is defined by a maintained correspondence between physical asset state and a decision model that improves a documented maintenance outcome while preserving traceability and human accountability.

Keywords *digital twin; predictive maintenance; condition monitoring; Industry 4.0; manufacturing SMEs; remaining useful life*

1 Introduction

Predictive maintenance seeks to identify developing degradation and schedule action before functional failure while avoiding unnecessary time based work. It combines condition observations, historical context, models, and maintenance planning. Manufacturing organizations value the approach because unplanned stoppage can disrupt production, quality, safety, and delivery. Implementation is difficult because failures are relatively rare, operating conditions change, maintenance records are incomplete, and a useful warning must arrive early enough to change a decision. Surveys of industrial predictive maintenance show that data quality, labels, model transfer, and integration with maintenance practice remain recurring challenges [1]-[3].

A digital twin can organize these elements around an asset or system. Early product lifecycle concepts linked physical and virtual spaces through information flow [4], and later definitions emphasized virtual information constructs, in use data, and simulation [5]. The term is now applied widely, sometimes to a static model or dashboard. Reviews distinguish a digital model with manual

exchange, a digital shadow with automated one way data flow, and a digital twin with automated exchange that can influence both representations [6]. Broader analyses describe the enabling technologies, manufacturing applications, data relationships, and design implications of this concept [10]-[13], [19]. For maintenance, the name is less important than the capabilities and decision loop that are actually implemented.

Small and medium enterprises face particular constraints. They may operate mixed age equipment from several vendors, with limited sensor coverage, unavailable interfaces, short engineering windows, and small maintenance teams. A platform designed for a new automated factory may be too expensive or difficult to sustain. A minimum viable approach should begin with one asset and failure mode, use existing measurements where possible, and add complexity only when it improves detection, diagnosis, prognosis, or planning. The ongoing cost of data stewardship and model maintenance belongs in the business case.

This article synthesizes digital twin and predictive maintenance research into an implementation framework for manufacturing SMEs. It is a conceptual technical review and does not present invented field results. The contribution is a capability model, architecture, method selection guide, data governance plan, validation strategy, and staged roadmap. The framework treats the maintenance decision as the organizing requirement and uses traceability to connect measurements, models, recommendations, completed work, and outcomes.

Existing digital twin reviews [6], [10]-[13], [19] establish definitions, enabling technologies, and application taxonomies across manufacturing broadly, and a recent requirement-based roadmap has begun to standardize predictive-maintenance automation functions using digital twin technology [18]. These works are largely scale-agnostic and do not address why a full-fidelity twin is frequently not viable for an SME with limited sensor coverage and a small maintenance team. This review's specific contribution is therefore not a new modeling technique but a minimum-viable sequencing of capabilities, tied to a single named maintenance decision at each stage, so that an SME can obtain decision value before undertaking platform-scale investment.

2 Review Methodology

This review synthesizes literature on predictive maintenance, digital twin definitions and architecture, condition monitoring standards, and physics-based and data-driven prognostics, retrieved from IEEE Xplore, ScienceDirect, and relevant ISO standards documents, using combinations of the terms *digital twin*, *predictive maintenance*, *condition monitoring*, *remaining useful life*, *digital shadow*, and *Industry 4.0*. ISO condition-monitoring and digital-twin framework standards were included as authoritative structural references regardless of publication date; systematic literature reviews and empirical studies were prioritized from the last ten years, with particular attention to sources published since 2020 to capture the maturing digital twin literature. Sources were included when they specified a reproducible method, evaluated architecture, or documented implementation constraint relevant to predictive maintenance; sources without methodological detail were excluded. Because the evidence spans standards, systematic reviews, and individual case implementations with different scopes, synthesis was narrative rather than statistical. Where general digital twin guidance assumed data or staffing resources unlikely to be available to an SME, the more conservative, resource-constrained position was adopted and expressed as a design requirement in Sections 4 and 7, consistent with the approach used in comparable technical-review articles in this journal.

3 Digital Twin Capabilities for Maintenance

A maintenance twin starts with persistent asset identity. The record links machine, component, configuration, location, controller program, sensor channels, maintenance history, and applicable documentation. Without configuration context, the same vibration or temperature value can have different meanings after a motor, bearing, tool, or operating recipe changes. Identity also prevents data from a retired or replaced component being treated as a continuous history. The twin should therefore record effective dates and relationships rather than store only current values.

The second capability is synchronized state observation. Raw sensor signals, controller tags, quality measurements, and production context require common time references and engineering units. The twin may store high frequency data locally and derived indicators centrally. It should preserve the transformation from raw measurements to features so that an alert can be reconstructed. Quality flags identify missing data, sensor saturation, calibration status, and communication delay. A visualization that hides these conditions can create false confidence even if it looks complete.

The third capability is state estimation and comparison. A reference can be a threshold, statistical baseline, learned representation, physics model, or combination. The twin compares observed behavior with expected behavior for the current load, speed, product, and environment. Context is essential because normal changes in operating point can resemble damage. A useful health indicator should be sensitive to the target degradation and relatively insensitive to irrelevant variability. It should also be understandable enough to support maintenance review.

The fourth capability is prediction or what if analysis. Prognostic models estimate future health or remaining useful life under an assumed usage profile. Simulation can compare continued operation, reduced load, inspection, or replacement. These outputs are conditional, not certainties. The twin should report prediction intervals, assumptions, data recency, and known model limits. A precise single failure date can mislead planners when future load and degradation mechanisms are uncertain. Decision support should translate uncertainty into options and consequences.

The fifth capability is feedback. Completed inspection, repair, replacement, and observed component condition are returned to the twin. This information confirms or corrects the earlier diagnosis and updates labels for future learning. If maintenance outcomes are not captured, the system can produce repeated alerts without learning whether they were useful. Feedback also measures operational value through avoided failures, reduced inspection, changed spare parts use, and production impact. The loop makes the twin a maintained engineering system rather than a one way monitoring dashboard.

Table 1 Digital twin maintenance capabilities

Capability	Minimum implementation	Decision value
Asset identity	Versioned component configuration location and history	Correctly associates evidence with the maintained item
State observation	Synchronized signals context and quality flags	Shows current condition with known data limitations
Health assessment	Reference behavior anomaly and diagnosis	Identifies whether investigation is needed
Prognosis	Conditional future health with uncertainty	Supports timing and resource planning
Feedback	Completed work findings and outcome	Measures usefulness and improves future models

4 Reference Architecture

Condition monitoring standards provide a useful functional structure. ISO 13374 describes processing and communication blocks for condition monitoring and diagnostics [7]. A digital twin implementation can map these into data acquisition, data manipulation, state detection, health assessment, prognostic assessment, advisory generation, and human interaction. This modularity separates measurement, analytics, and decision support. It also permits an SME to begin with acquisition and state detection, then add prognosis only when data and decisions justify it.

The physical layer includes the asset, sensors, controller, and existing protection. Data acquisition collects vibration, current, temperature, pressure, speed, load, cycle count, alarms, and production context as relevant to the selected failure mode. The integration layer normalizes units, timestamps events, checks quality, and associates measurements with asset identity and configuration. The model layer estimates state and health. The service layer creates alerts, forecasts, and maintenance options. The workflow layer connects these outputs with planning, work orders, spare parts, and operator feedback.

A bidirectional connection should not imply unrestricted automatic control. For many predictive maintenance cases, the twin influences the physical system through an approved work order, inspection, or operator action. Direct changes to speed or setpoints require a separate control safety assessment. The interface can begin as advisory and later automate low consequence tasks such as

requesting a repeat measurement or scheduling a nonintrusive inspection. Authority should expand only after false alerts, missed detections, and workflow impact are understood.

Interoperability becomes important when the first pilot scales. Asset information, sensor semantics, model versions, and maintenance events should use stable identifiers and documented schemas. ISO 23247 provides a digital twin framework for manufacturing that can guide domain, entity, communication, and functional views [8]. The organization does not need to implement every element at once, but adopting consistent names and interfaces early reduces later data reconciliation. Vendor platforms should support export of measurements, configuration, and model outputs.

Edge and central computing have complementary roles. Edge processing supports high frequency feature extraction, low latency alerts, and operation during network loss. Central services support history, model training, cross asset comparison, and maintenance planning. The system should define what happens when either layer fails. Local equipment protection must remain independent. Buffered measurements can synchronize after connectivity returns, while stale central recommendations should expire. This behavior should be tested rather than assumed.

5 Modeling Options

Threshold and rule based methods are appropriate when a physical limit or established condition indicator exists. They need little failure data, are easy to explain, and provide a strong baseline. Their main weakness is sensitivity to operating context and the difficulty of combining several weak indicators. Statistical process monitoring can account for normal variability and identify multivariate deviation. These methods often deliver earlier value than complex learning because they expose data and workflow problems without demanding a large labeled history.

Data driven models learn relationships between features and health, fault class, or remaining life. Classification supports fault identification, regression estimates a continuous health measure, and unsupervised methods detect departure from normal behavior. Machine learning reviews report wide use of support vector machines, tree ensembles, neural networks, and deep architectures in predictive maintenance [1]-[3]. Model choice should follow data volume, signal structure, interpretability, update needs, and the cost of error. Deep learning is not automatically appropriate for a small dataset collected under narrow conditions.

Physics based models represent degradation, loads, dynamics, or failure mechanisms. They can extrapolate beyond observed failures and support what if simulation, but require domain knowledge and parameter calibration. For an SME, a reduced model may be more sustainable than a high fidelity multiphysics model. Parameters can be updated from measurements, and the residual between predicted and observed behavior becomes a condition feature. The model should be no more complex than the decision requires.

Hybrid models combine physical structure with data driven correction. A physics model can constrain a learning algorithm, generate simulated cases, or provide features. A learning model can estimate uncertain parameters or residual error. Component based reduced order models support efficient predictive twins while preserving interpretable relationships [9], and machine tool implementations illustrate how hybrid maintenance models can be connected to a twin [16]. Hybrid approaches are attractive when failure data are scarce, but they still require validation of both the physical assumptions and the learned component. Combining two uncertain models does not guarantee a reliable result.

Model selection should account for consequence and maintenance action. If the response is a low cost inspection, a sensitive anomaly detector with some false positives may be acceptable. If the recommendation is to stop a critical line, evidence and explanation requirements are higher. Remaining useful life is useful only when uncertainty is narrow enough to change scheduling. Otherwise a health trend and inspection trigger may be more honest. The framework therefore links each output to a decision, lead time, cost of false alarm, and cost of missed detection.

Table 2 Model selection by evidence and purpose

Model type	Data need	Main strength	Main limitation
Threshold and rules	Engineering limit and normal range	Transparent and fast to commission	Weak adaptation to context

Model type	Data need	Main strength	Main limitation
Statistical model	Representative normal and event history	Quantifies multivariate deviation	Assumes stable distributions
Machine learning	Representative labeled or normal data	Captures nonlinear signal relationships	Drift labels and explainability
Physics model	Mechanism parameters and operating inputs	Supports extrapolation and what if analysis	Calibration and modeling effort
Hybrid model	Physical structure plus operational data	Balances constraints and adaptation	Validates two interacting model sources

6 Data and Model Lifecycle

Data readiness is evaluated for a specific failure mode. The team identifies the physical precursor, available sensors, sample rate, operating context, historical duration, maintenance labels, and expected warning horizon. Measurements need sufficient sensitivity and placement. A model cannot recover information that the sensing system does not capture. Pilot instrumentation should therefore be justified by a failure mode and effects analysis, not by a general desire for more data. Temporary high quality measurements can help determine whether a lower cost permanent sensor is adequate. Maintenance records require structured feedback. Each work event should identify symptom, diagnosis, action, component, observed condition, parts used, downtime, and return to service. Free text can remain, but key fields support learning and evaluation. Time alignment between alerts and work is essential. A component replacement resets part of the degradation history, while a lubrication task changes expected behavior differently. Configuration version and operating hours since intervention should be available to the model.

Training and testing use chronological separation and, where possible, asset separation. Randomly mixing adjacent windows from the same run can create severe leakage because samples are highly correlated. Evaluation should compare a naive threshold or current maintenance rule. Metrics include detection lead time, false alerts per operating period, missed events, diagnosis accuracy, calibration of uncertainty, and the effect on maintenance decisions. Class accuracy alone is misleading when healthy data dominate.

Model governance records training data version, feature code, algorithm, parameters, performance, intended assets, excluded conditions, owner, approval, and deployment date. Drift monitoring compares current feature and output distributions with the approved range. A model is not retrained automatically simply because a statistical test changes. Engineers first investigate sensor replacement, operating mix, maintenance, and configuration. Retraining follows the same validation and approval as the original model. The previous version remains available for rollback.

The twin itself needs maintenance. Sensors require calibration or replacement, gateways receive patches, schemas change, and staff responsibilities move. Maintenance reviews and deployed condition monitoring platforms show why integration and lifecycle work must be treated as continuing operational responsibilities [15], [17]. Annual cost should include data storage, connectivity, licenses, model review, false alert investigation, and integration support. A pilot that depends on a researcher manually correcting data is not yet an operational system. Readiness includes named ownership and a budget for lifecycle tasks, not only a successful demonstration.

7 Validation and Value Measurement

Validation proceeds from component to decision. Sensor tests confirm range, noise, timing, and behavior under known conditions. Data pipeline tests verify units, timestamps, missing value handling, and asset association. Model tests use historical and injected scenarios. Service tests confirm alert delivery, acknowledgement, and expiration. Workflow tests verify that planners can create an appropriate inspection or work order. End to end exercises measure the time from physical indication to completed decision and feedback.

Failure data scarcity requires careful evidence. Run to failure experiments may be impractical, so teams can use seeded faults on a test asset, accelerated component tests, physics simulation, maintenance findings, and expert review. These sources are not interchangeable and should be labeled. Synthetic data can test software paths and rare conditions, but reported performance must distinguish synthetic from field observations. A model trained on simulation should be evaluated for

domain difference before production use. A requirement based roadmap can keep this evidence tied to standardized predictive maintenance functions as maturity increases [18].

Shadow operation is a strong intermediate stage. The twin generates health assessments and recommendations while the existing maintenance process remains authoritative. Engineers review alerts, record whether an action would have changed, and identify missing context. Shadow results reveal alert frequency and workflow burden without risking production. The period must be long enough to cover relevant operating regimes. Absence of failures during the period does not prove detection performance, but it can demonstrate data stability and manageable false alerts.

Value measurement compares the new process with a defined baseline. Possible outcomes include fewer emergency work orders, longer warning time, reduced unnecessary inspection, lower secondary damage, improved spare parts readiness, reduced downtime, or better quality. Digital twin driven prognostics provides a technical basis for connecting health assessment with these maintenance decisions [14]. Financial estimates should show assumptions and avoid counting the same benefit twice. The organization should also report added labor and technology cost. A twin that reduces one failure but creates persistent analysis workload may not scale.

Scaling decisions should be based on repeated patterns, not a platform mandate. Assets with similar failure mechanisms, sensors, and operating context are good candidates for reuse. Transfer to different equipment requires recalibration and validation. A reusable data schema and deployment process may provide more value than reusing the same predictive model. Portfolio planning can rank assets by consequence, failure frequency, detectability, data readiness, and expected decision value.

8 Implementation Roadmap

The first phase defines one decision. Examples are whether to inspect a gearbox within the next shift, schedule a bearing replacement during the next planned stop, or continue a pump under reduced load. The team documents the failure mode, consequence, current rule, lead time, available signals, and person who acts. A project without a named decision often becomes a dashboard whose use is difficult to evaluate.

The second phase establishes a data foundation and baseline. Asset identity, configuration, sensors, context, and maintenance events are connected. Engineers inspect raw signals and create simple thresholds or statistical indicators. This phase tests whether the failure precursor is observable and whether the workflow can use an alert. Data quality targets include availability, timestamp alignment, stable units, and known sensor health. If these fail, model development pauses while instrumentation is corrected.

The third phase builds and validates a candidate model. The team compares rule based, statistical, data driven, physics based, and hybrid options according to evidence and consequence. Testing uses chronological splits, realistic class imbalance, and decision metrics. The model runs in shadow mode, and each alert receives a disposition. Acceptance requires not only predictive improvement but also an understandable recommendation, controlled update process, and defined fallback.

The fourth phase integrates maintenance planning. Approved outputs create a review task or work request with asset, evidence, urgency, confidence, and suggested next action. Planners retain authority unless direct automation has a separate safety case. Completed work returns findings to the twin. The team measures response time, useful alerts, false alert burden, and changed maintenance outcomes. These measures determine whether to continue, revise, or stop the pilot.

The fifth phase standardizes successful patterns. The organization defines reusable sensor packages, naming, data contracts, model templates, cybersecurity controls, validation tests, and support responsibilities. Scaling proceeds to assets with comparable failure modes and value. Governance reviews model drift, exceptions, and realized benefit at a fixed interval. This staged process limits sunk cost and builds internal capability while maintaining a direct connection between technology and maintenance outcomes.

Table 3 Implementation gates for an SME pilot

Gate	Question	Required evidence
Decision	Will the output change a named maintenance action	Owner lead time consequence and baseline process
Observation	Is the degradation precursor measurable	Sensor tests context and data quality
Model	Does the candidate outperform a simple baseline	Chronological evaluation uncertainty and failure tests
Workflow	Can staff act and return findings	Shadow operation alert disposition and work integration
Scale	Will benefit exceed lifecycle cost on similar assets	Realized outcomes support model and ownership plan

9 Limitations

This review is narrative and design-oriented rather than a systematic review with a registered protocol, and it does not claim exhaustive retrieval of the digital twin and predictive maintenance literature or a formal meta-analysis; the heterogeneity of asset types, failure modes, and evaluation metrics across the cited sources limits direct quantitative comparison. The proposed minimum-viable capability sequence and implementation gates are conceptual and have not been validated on an instrumented SME pilot; quantitative results attributed to individual cited studies describe those studies' own reported outcomes and should not be read as evidence already obtained for the framework proposed here. Because manufacturing SMEs differ substantially in equipment age, sensor availability, and maintenance staffing, the cost and duration of following the proposed roadmap will vary by site and are not quantified in this review.

10 Conclusion

A manufacturing digital twin supports predictive maintenance when it maintains a traceable relationship among physical asset state, configuration, models, decisions, and maintenance feedback. Visual replication and large data volume are secondary. The proposed framework organizes capabilities according to asset identity, synchronized observation, state assessment, prognosis, advisory generation, human interaction, and feedback. It allows SMEs to begin with thresholds and a digital shadow, then adopt data driven, physics based, or hybrid models when the decision and evidence justify them. Validation must cover the entire path from sensor to work outcome, and lifecycle cost must include maintenance of the twin. A staged minimum viable implementation offers a more credible route to value than a broad platform first program.

AUTHOR CONTRIBUTIONS

CRediT: conceptualisation, literature synthesis, framework development, writing, review, and approval of the manuscript.

FUNDING

The authors declare that this research received no funding from any source.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

DATA AVAILABILITY STATEMENT

No original datasets were generated or analyzed in this study, as this article presents a conceptual technical review and design framework.

References

- [1] W. Zhang, D. Yang, and H. Wang, Data driven methods for predictive maintenance of industrial equipment a survey, *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213-2227, 2019. <https://doi.org/10.1109/JSYST.2019.2905565>

- [2] T. P. Carvalho et al., A systematic literature review of machine learning methods applied to predictive maintenance, *Computers and Industrial Engineering*, vol. 137, article 106024, 2019. <https://doi.org/10.1016/j.cie.2019.106024>
- [3] T. Zonta et al., Predictive maintenance in the Industry 4.0 a systematic literature review, *Computers and Industrial Engineering*, vol. 150, article 106889, 2020. <https://doi.org/10.1016/j.cie.2020.106889>
- [4] M. W. Grieves, Product lifecycle management the new paradigm for enterprises, *International Journal of Product Development*, vol. 2, nos. 1-2, pp. 71-84, 2005. <https://doi.org/10.1504/IJPD.2005.006669>
- [5] M. Grieves and J. Vickers, Digital Twin mitigating unpredictable undesirable emergent behavior in complex systems, in *Transdisciplinary Perspectives on Complex Systems*, Springer, pp. 85-113, 2017. https://doi.org/10.1007/978-3-319-38756-7_4
- [6] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, Digital Twin in manufacturing a categorical literature review and classification, *IFAC PapersOnLine*, vol. 51, no. 11, pp. 1016-1022, 2018. <https://doi.org/10.1016/j.ifacol.2018.08.474>
- [7] ISO, Condition monitoring and diagnostics of machines Data processing communication and presentation Part 1 General guidelines, ISO 13374-1, 2003.
- [8] ISO, Automation systems and integration Digital twin framework for manufacturing Part 1 Overview and general principles, ISO 23247-1, 2021.
- [9] M. G. Kapteyn, D. J. Knezevic, D. Huynh, M. Tran, and K. Willcox, Data driven physics based digital twins via a library of component based reduced order models, *International Journal for Numerical Methods in Engineering*, vol. 123, no. 13, pp. 2986-3003, 2022. <https://doi.org/10.1002/nme.6423>
- [10] A. Fuller, Z. Fan, C. Day, and C. Barlow, Digital Twin enabling technologies challenges and open research, *IEEE Access*, vol. 8, pp. 108952-108971, 2020. <https://doi.org/10.1109/ACCESS.2020.2998358>
- [11] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, Characterising the Digital Twin a systematic literature review, *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36-52, 2020. <https://doi.org/10.1016/j.cirpj.2020.02.002>
- [12] M. Liu, S. Fang, H. Dong, and C. Xu, Review of digital twin about concepts technologies and industrial applications, *Journal of Manufacturing Systems*, vol. 58, pp. 346-361, 2021. <https://doi.org/10.1016/j.jmsy.2020.06.017>
- [13] Q. Qi and F. Tao, Digital Twin and Big Data towards smart manufacturing and Industry 4.0 a 360 degree comparison, *IEEE Access*, vol. 6, pp. 3585-3593, 2018. <https://doi.org/10.1109/ACCESS.2018.2793265>
- [14] F. Tao, M. Zhang, Y. Liu, and A. Nee, Digital twin driven prognostics and health management for complex equipment, *CIRP Annals*, vol. 67, no. 1, pp. 169-172, 2018. <https://doi.org/10.1016/j.cirp.2018.04.055>
- [15] A. Errandonea, S. Beltran, and S. Arrizabalaga, Digital Twin for maintenance a literature review, *Computers in Industry*, vol. 123, article 103316, 2020. <https://doi.org/10.1016/j.compind.2020.103316>
- [16] W. Luo, T. Hu, Y. Ye, C. Zhang, and Y. Wei, A hybrid predictive maintenance approach for CNC machine tool driven by Digital Twin, *Robotics and Computer Integrated Manufacturing*, vol. 65, article 101974, 2020. <https://doi.org/10.1016/j.rcim.2020.101974>
- [17] M. Short and J. Twiddle, An industrial digitalization platform for condition monitoring and predictive maintenance of pumping equipment, *Sensors*, vol. 19, no. 17, article 3781, 2019. <https://doi.org/10.3390/s19173781>
- [18] S. Ma, K. A. Flanigan, and M. Berges, State of the art review and synthesis a requirement based roadmap for standardized predictive maintenance automation using digital twin technologies, *Advanced Engineering Informatics*, vol. 61, article 102800, 2024. <https://doi.org/10.1016/j.aei.2024.102800>

- [19] B. R. Barricelli, E. Casiraghi, and D. Fogli, A survey on Digital Twin definitions characteristics applications and design implications, IEEE Access, vol. 7, pp. 167653-167671, 2019. <https://doi.org/10.1109/ACCESS.2019.2953499>

© 2026 The Author(s) | Open Access