



AI-Driven Project Management Automation: A Framework for Task Prioritization, Progress Monitoring, and Risk Identification

Sujata Manohar Borkar

School of Business, Indira University, Pune

How to Cite this Article:

Borkar, S. M. (2026). AI-Driven Project Management Automation: A Framework for Task Prioritization, Progress Monitoring, and Risk Identification. International Journal of Creative and Open Research in Engineering and Management, 2(10), 1-9.
<https://doi.org/10.55041/ijcope.v2i10.134>

License:

This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.



<https://doi.org/10.55041/ijcope.v2i10.134>

Abstract

Project management requires coordination of tasks, deadlines, dependencies, resources, and risks. Manual status reporting and inconsistent prioritisation can make emerging delivery problems difficult to identify. This paper proposes and demonstrates a framework connecting task prioritisation, progress monitoring, and task-delay risk identification. The framework combines transparent rule-based scoring, monitoring flags, a machine-learning classifier, and human review. A prototype was evaluated using 1,000 generated task records, with a Decision Tree compared against a majority-class baseline on a held-out test set. The Decision Tree achieved 61.0% accuracy, 27.4% precision, 57.5% recall, and an F1-score of 37.1% for the delay class. The baseline achieved 80.0% accuracy but detected no delayed tasks, demonstrating why accuracy alone can be misleading in imbalanced classification. These results describe the prototype experiment and do not establish effectiveness on real project records. The paper concludes that an integrated framework can provide a reproducible basis for further evaluation, with real-world validation required before operational use.

Keywords: artificial intelligence, project management automation, task prioritisation, progress monitoring, risk identification, delay prediction, Decision Tree, business analytics

1. Introduction

Project management connects business objectives with the tasks, people, schedules, and resources required to deliver them. Across software, consulting, operations, and other project-based environments, managers must coordinate dependencies, allocate limited capacity, monitor deadlines, and respond to changing requirements. These responsibilities become more difficult when information is spread across spreadsheets, meetings, email updates, and project-management platforms.

Manual status reporting remains useful for context and accountability, but it can create delays between a problem emerging and a manager noticing it. Task owners may update progress at different times, risk descriptions may be inconsistent, and competing priorities can make it difficult to decide which task requires attention first. A consolidated analytical workflow can help organise information and identify conditions that warrant review.

Artificial intelligence (AI) and analytics offer several possible forms of support, including classification, pattern recognition, forecasting, and recommendation. However, it is important to distinguish conventional automation from AI. A rule that flags an overdue task is rule-based workflow automation. A machine-learning model that estimates the likelihood of delay from task attributes is predictive analytics. Both can be useful, but they have different data requirements and should be evaluated separately.

This paper proposes a framework linking task prioritisation, progress monitoring, delay-risk identification, reporting, and human review. A prototype illustrates the workflow using generated task records and evaluates a Decision Tree classifier against a simple majority-class baseline. The purpose is to demonstrate a reproducible method and identify



what must be tested before real-world use, rather than to claim that the framework has already improved project outcomes.

1.1 Problem Statement

Project teams often have to decide which tasks need immediate attention while balancing deadlines, business importance, dependencies, resource availability, and uncertainty. If prioritisation depends only on subjective judgment, teams may apply criteria inconsistently. If monitoring focuses only on overdue tasks, managers may miss tasks that are not yet late but have several risk signals. Predictive models may help identify patterns, but only when data are sufficiently reliable and model performance is evaluated against meaningful baselines.

The research problem is therefore how to combine transparent priority scoring and rule-based monitoring with a tested predictive component in a single decision-support framework, while retaining human accountability and acknowledging the limitations of generated data.

1.2 Research Aim and Objectives

The aim of this study is to design and demonstrate a framework for task prioritisation, progress monitoring, and task-delay risk identification using transparent rules and a baseline machine-learning model.

1. To design a transparent task-prioritisation approach using urgency, business importance, dependencies, risk indicators, and resource readiness.
2. To define an automated progress-monitoring approach that identifies overdue, blocked, or deadline-sensitive tasks.
3. To implement and evaluate a baseline machine-learning model for identifying task-delay risk.

1.3 Research Questions

4. How can task attributes be combined into a transparent and repeatable prioritisation score?
5. How can rule-based monitoring identify tasks that require timely review?
6. What does a baseline Decision Tree reveal about delay classification on generated task records, and why should its results be interpreted cautiously?

2. Literature Review

2.1 Artificial Intelligence in Project Management

AI can support project management through prediction, classification, pattern recognition, and decision support. Taboada et al. (2023) review AI-enabled project-management research and identify applications across project-management activities. Davenport and Harris (2007) provide a broader rationale for using analytical decision-making in organisations. These perspectives suggest that the value of AI is not simply automation for its own sake, but the ability to use information consistently to support a decision.

AI outputs are only as useful as the information and assumptions behind them. Project plans change, tasks can be redefined, and delay outcomes may be influenced by factors that are not recorded in a dataset. Therefore, a model should be understood as a source of evidence for review, rather than an unquestionable authority. Model design, data quality, evaluation strategy, and the intended use of predictions all affect whether a system is appropriate.

2.2 Task Prioritisation and Automation

Task prioritisation is a decision process for ordering work according to the consequences of delay, deadline urgency, dependencies, effort, business importance, risk, and available resources. A weighted score makes these criteria explicit and allows teams to apply the same decision logic across tasks. It also makes the method easier to explain than an opaque score. Nevertheless, the selected weights reflect design assumptions and should be reviewed with project stakeholders.



Many recurring project actions can be automated without machine learning. Examples include sending reminders, flagging blocked tasks, highlighting deadlines approaching within a defined window, and preparing routine status summaries. Such rules can improve consistency but do not learn from historical examples. Predictive modelling is a separate layer that estimates outcomes from labelled data and must be tested to establish whether it provides useful information beyond the rules.

2.3 Progress Monitoring and Risk Identification

Progress monitoring compares task status, milestones, and expected outcomes with the project plan. Indicators may include overdue status, blocked dependencies, milestone slippage, effort variance, and time since the last update. Rule-based monitoring detects conditions defined in advance; a supervised model estimates an outcome from examples where the outcome label is known.

Evaluation metrics should reflect the operational problem. Accuracy measures the proportion of all predictions that are correct, but it can be misleading when one outcome class is much more common than another. Precision indicates how many predicted delay cases were actually labelled as delays. Recall indicates how many labelled delay cases were detected. The F1-score combines precision and recall into one measure. A confusion matrix shows the counts of correct and incorrect predictions for each class.

2.4 Human–AI Collaboration

Project management involves negotiation, stakeholder communication, changing requirements, and accountability. An automated alert may lack context about a client decision, an approved scope change, or a dependency outside the team's control. Human review is therefore essential when interpreting alerts, deciding whether a risk is material, and selecting corrective action. The framework positions AI as decision support and leaves consequential decisions with responsible project managers.

2.5 Research Gap

Task prioritisation, progress monitoring, workflow automation, and risk management are often described as separate functions. This paper addresses a design-oriented challenge by connecting these functions in a single transparent workflow and distinguishing rule-based automation from predictive modelling. Its contribution is a reproducible prototype design and a documented evaluation on generated data; it does not claim that commercial project-management systems lack similar capabilities.

3. Conceptual Framework

The proposed framework has five connected layers. The first is the task-data layer, which stores task characteristics such as deadlines, importance, dependencies, resource load, estimated effort, team experience, requirements changes, and blocked status. The second is the prioritisation layer, which converts selected attributes into a score. The third is the monitoring layer, which applies explicit rules to identify blocked or deadline-sensitive tasks. The fourth is the predictive layer, where a classifier estimates delay risk from task attributes. The fifth is the decision layer, where project managers review the score, flags, and prediction before taking action.

- Task-data layer: organise task attributes in a consistent structured format.
- Prioritisation layer: calculate a weighted score from 0 to 100.
- Monitoring layer: flag blocked tasks and tasks with three or fewer days remaining before the deadline.
- Prediction layer: classify delay risk using a Decision Tree model trained on labelled examples.
- Human-review layer: investigate alerts, assess context, and decide on an appropriate response.

The layers are intentionally separable. A team can use the prioritisation and monitoring rules before adopting a predictive model. If a model does not outperform a simple baseline or produces too many false alarms, it should not automatically be deployed. This modular approach makes it possible to evaluate each component independently.



4. Research Methodology

4.1 Research Design

The study uses a design-oriented prototype approach. The work includes framework development, generation and preparation of structured task records, implementation of a weighted priority score and monitoring rules, model training, and evaluation against a baseline. The experiment is intended to test the technical workflow within a defined data-generating process.

4.2 Data and Variables

The prototype contains 1,000 generated task records. Variables include days to deadline, business importance, dependency count, resource-load ratio, estimated effort, team experience, a requirements-change indicator, a blocked-task indicator, and a binary delay label. The data were created for demonstration and do not represent actual organisational records. A fixed random seed supports reproducibility.

The delay label was generated using an illustrative probability formula and random sampling. This is a central limitation: the model is learning patterns in an artificial label-generation process. The experiment does not establish that the same relationships exist in real projects or that the model would achieve similar performance on actual project records.

4.3 Task Prioritisation and Monitoring

The priority score ranges from 0 to 100 and combines five normalised factors: deadline urgency (25%), business importance (25%), dependency impact (20%), risk indicators (20%), and resource readiness (10%). These weights are illustrative design choices and have not been validated by project stakeholders. The score provides a consistent ordering mechanism, but it should not be treated as an objective measurement of task value.

The monitoring logic flags tasks that are blocked or have three or fewer days remaining before the deadline. These flags are rule-based and are not machine-learning predictions. The rules are transparent and can be adjusted after consultation with intended users. In an operational setting, the organisation should define how deadlines, blocked status, and resource readiness are recorded so that flags are reliable.

4.4 Predictive Model and Evaluation

A Decision Tree classifier was trained using eight task attributes. Task ID, the delay label, the calculated priority score, and the monitoring flag were excluded from model inputs. The dataset was split into 80% training and 20% test records using stratification and a fixed random state. The Decision Tree used a maximum depth of five, a minimum leaf size of ten, balanced class weights, and a fixed random state. A majority-class DummyClassifier served as the baseline.

Evaluation measures included accuracy, precision and recall for the delay class, F1-score, and the confusion matrix. Using a held-out test set provides a basic check on records not used to fit the model. However, one train-test split does not show how stable results would be across different samples or future time periods. Repeated validation and time-aware testing should be considered before operational deployment.

4.5 Ethical Considerations

The prototype does not use identifiable employee records or confidential company data. Any future organisational evaluation would require appropriate authorisation, access controls, data minimisation, and careful interpretation of workload indicators. Task-delay predictions should not be used to rank employees or make consequential personnel decisions without suitable evidence, governance, and human review. A responsible implementation should document the purpose of the model, data access rules, known limitations, and the process for challenging or correcting an alert.

5. Findings and Discussion

5.1 Task Prioritisation and Monitoring

The prototype calculates a priority score for each task and creates a monitoring flag for blocked tasks or tasks approaching their deadline. These outputs provide a consistent way to organise tasks for review. A score can help



teams discuss why a task appears urgent, while a monitoring flag can prompt a timely check. However, the score and thresholds remain illustrative and should be evaluated with intended users before operational use.

A useful operational process would not simply sort tasks by score and follow the list mechanically. Managers should review whether a high-priority task is genuinely on the critical path, whether a dependency has changed, and whether the available resource has the required skills. The framework is most useful when it structures discussion and draws attention to tasks that might otherwise be overlooked.

5.2 Delay Classification Results

Model	Accuracy	Precision (delay)	Recall (delay)	F1 (delay)
Decision Tree	61.0%	27.4%	57.5%	37.1%
Majority-class baseline	80.0%	0.0%	0.0%	0.0%

On the held-out test set of 200 records, the Decision Tree achieved 61.0% accuracy, 27.4% precision, 57.5% recall, and an F1-score of 37.1% for the delay class. Its confusion matrix was $\begin{bmatrix} 99 & 61 \\ 17 & 23 \end{bmatrix}$, with rows representing actual no-delay and delay classes and columns representing predicted no-delay and delay classes.

The majority-class baseline achieved 80.0% accuracy by predicting no delay for every task. It did not identify any delayed tasks. The Decision Tree detected 23 of the 40 delayed tasks in the test set, but it also produced false-positive alerts. In this experiment, the model detected more delayed cases than the baseline, while the baseline's higher accuracy reflected its tendency to predict the majority class.

5.3 Interpretation of Model Metrics

The results demonstrate why a model should not be selected on accuracy alone. If the operational goal is to identify tasks at risk of delay, a model that predicts no delays can have apparently high accuracy while providing no delay detection. Recall matters because missed delays may reduce the opportunity for early intervention. Precision also matters because false-positive alerts consume managerial attention and can create alert fatigue. The appropriate balance depends on the consequences of missed risks and unnecessary reviews.

The Decision Tree's delay precision of 27.4% indicates that many tasks flagged as delayed were not labelled as delayed in the test set. Its recall of 57.5% indicates that it identified a little over half of the labelled delay cases. These results are not strong evidence for production deployment. They are useful as a baseline experiment that reveals the need for better data, more careful validation, and potentially improved features or modelling approaches.

5.4 Limits of Interpretation

The measured performance is specific to the generated records and illustrative delay-label formula. It should not be interpreted as evidence of real-world predictive accuracy, causal relationships, or reduced project delays. Because the label was generated from a formula, the model may learn patterns built into that formula rather than relationships that hold in operational project environments. A genuine field evaluation would require authorised historical records and an independently defined delay outcome.

6. Discussion: Implications for Project Management

An integrated workflow can make task priorities, progress indicators, and potential risks easier to review in one place. Transparent rules provide a practical starting point because they are easy to explain and adjust. Predictive models can be added when reliable labelled data are available and when evaluation shows that they contribute useful information beyond existing rules.

The first managerial implication is the importance of clear task data. Deadlines, task status, dependencies, effort estimates, and change indicators need consistent definitions. Missing or stale information can lead to misleading scores and alerts. The second implication is that monitoring should be connected to a response process. An alert has limited value if nobody owns the review, if the reason cannot be understood, or if there is no agreed escalation path.



The third implication concerns model governance. Project managers should be able to inspect why a task was flagged, correct inaccurate task information, and record whether an alert was useful. A model should be monitored for false positives and false negatives after deployment, because project types, working practices, and delivery conditions can change. Any claimed benefit—such as reduced reporting time, lower delay rates, or improved resource allocation—must be measured through a future comparison with current practices.

A future real-world study should compare model performance with an operational rule-based baseline and use records from projects or time periods not used during model training. It should also assess whether the predictions arrive early enough to support action, whether users understand them, and whether the benefits justify the cost of maintaining the system.

7. Limitations of the Study

- The generated records and labels may not represent the complexity, data quality, or changing conditions of actual projects.
- The delay label was created using an illustrative formula, so performance does not establish external validity.
- The evaluation uses one train-test split; repeated validation or time-aware testing may produce different results.
- The priority weights and monitoring thresholds are illustrative and have not been validated with project stakeholders.
- No production integration, user study, reporting-time experiment, or real-world cost-benefit analysis was conducted.
- The model does not establish causal relationships between task attributes and delay outcomes.

These limitations define the boundary of the claims that can be made. The paper demonstrates a prototype and its behaviour on generated records; it does not demonstrate organisational effectiveness. Future research should address these limitations in a staged way rather than moving directly from a prototype to operational deployment.

8. Recommendations and Future Research

8.1 Validate with authorised project data

The next stage should use appropriately authorised historical project records with a clear, independently defined delay outcome. Data collection should include task creation date, planned and actual completion dates, status history, dependency information, estimated effort, changes in scope, and relevant resource-capacity measures. Data access should be limited to what is necessary for the research.

8.2 Compare against meaningful baselines

Future evaluation should compare the Decision Tree with simple rule-based approaches and suitable alternative models. A more complex model should only be preferred if it provides measurable improvement on metrics relevant to the project-management use case. Evaluation should consider precision, recall, F1-score, confusion matrices, calibration where appropriate, and the timing of predictions.

8.3 Validate priority weights and thresholds

Project managers and task owners should review the five prioritisation factors and their weights. Workshops or structured feedback can identify whether the score reflects real decision needs. The deadline threshold for monitoring should also be tested across project types and task durations. Changes should be documented so the score remains interpretable.

8.4 Measure operational impact

A field pilot should measure whether the framework changes review practices or outcomes. Potential measures include time required to prepare status reports, proportion of alerts reviewed, proportion of alerts judged useful, lead time between an alert and a delay, missed-delay rate, and user confidence in the system. These measures should be



compared with a baseline period or similar projects. Savings or performance improvements should only be claimed if observed and documented.

8.5 Maintain human oversight

The system should support—not replace—project-management judgment. Every consequential alert should have an identifiable reason and a responsible reviewer. Teams should be able to correct source data, record alert outcomes, and escalate uncertain cases. This process will help improve trust and provide feedback for future model evaluation.

9. Conclusion

This paper presents a framework connecting task prioritisation, progress monitoring, delay-risk identification, reporting, and human review. The prototype demonstrates a weighted task score, rule-based monitoring flags, and a Decision Tree classifier evaluated against a majority-class baseline. Although the Decision Tree detected some labelled delays, the experiment also produced false alarms and cannot establish performance on actual project records.

The main contribution is the integration of transparent rules and a baseline predictive component in a reproducible workflow, alongside a clear discussion of evaluation and governance requirements. The results show that accuracy alone can be misleading when a model is evaluated on an imbalanced classification problem. They also show why generated-data experiments must be described carefully and not presented as real-world validation.

The practical value of the framework remains a question for further research. Future work should evaluate the approach using authorised historical project data, apply time-aware validation where appropriate, compare additional models with transparent baselines, validate scoring criteria with project managers, and measure usability and operational impact. AI is best positioned here as a decision-support capability that complements human accountability.

References

- Davenport, T. H., & Harris, J. G. (2007). *Competing on analytics: The new science of winning*. Harvard Business School Press.
- Müller, R., Locatelli, G., Holzmann, V., Nilsson, M., & Sagay, T. (2024). Artificial intelligence and project management: Empirical overview, state of the art, and guidelines for future research. *Project Management Journal*, 55(1), 9–15. <https://doi.org/10.1177/87569728231225198>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., VanderPlas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Project Management Institute. (2021). *A guide to the project management body of knowledge (PMBOK Guide) (7th ed.)*. Project Management Institute.
- Taboada, I., Daneshpajouh, A., Toledo, N., & de Vass, T. (2023). Artificial intelligence enabled project management: A systematic literature review. *Applied Sciences*, 13(8), 5014. <https://doi.org/10.3390/app13085014>