

POLITECNICO DI TORINO
MSc in Engineering and Management



**Evaluating Large Language Models on Project
Management and Business Analysis Processes: A
Methodological Framework for Automated
Benchmarking**

Supervisor

Prof. Filippo Maria Ottaviani

Candidate

Salvatore Fiscale

s328135

Academic Year 2025/2026

*Ai miei genitori,
per la forza e il sostegno che non mi hanno mai fatto mancare.*

Abstract

The rapid adoption of large language models (LLMs) in knowledge-intensive professions has outpaced the development of rigorous methods for assessing their performance on domain-specific tasks. While general-purpose benchmarks abound, the project management and business analysis disciplines lack a structured, reproducible means of measuring how well such models support core professional processes. This thesis addresses that gap by designing, validating, and executing a methodological framework for the automated benchmarking of LLMs across project management and business analysis activities. The framework is grounded in a practitioner survey that identified and validated seven macro-processes central to the two disciplines. From these processes, twelve operational tasks were derived, spanning estimation, scheduling, dependency identification, risk and stakeholder analysis, requirements elicitation, and related activities. A second survey established the measurement validity of twenty-four evaluation criteria, organised around three trade-off families and mapped to the twelve tasks. Cognitive coverage across the task set is characterised through an established taxonomy of learning objectives, ensuring that the benchmark spans application, analysis, and evaluation levels of reasoning rather than recall alone. To enable reproducible measurement, the framework is implemented as a deterministic scoring pipeline that compares model outputs against curated ground-truth references using transparent, task-appropriate metrics. An item bank of 998 benchmark instances, weighted by process relevance, supports administration under controlled prompting configurations. The methodology is demonstrated through a full empirical benchmarking exercise on two publicly available LLMs evaluated under identical conditions, covering the complete workflow from response collection through automated scoring to comparative reporting; the two models obtained markedly different overall scores (0.807 and 0.545 respectively), with the performance gap concentrated in the project management domain. The contributions of the thesis include a survey-validated mapping from professional processes to measurable benchmark tasks; a transparent and extensible evaluation methodology suited to disciplines where ground truth is partly judgemental; and a reproducible implementation and executed comparison that allow additional models to be assessed under identical conditions. The framework offers practitioners and researchers a principled basis for evaluating the fitness of language models for project management and business analysis work, and provides a foundation for ongoing comparative assessment as models continue to evolve.

Contents

1	Introduction	13
1.1	Background and Motivation	13
1.2	Research Problem	15
1.3	Research Questions	16
1.4	Scope and Boundaries	17
1.5	Contributions	18
1.6	Thesis Structure	20
2	Literature Review	23
2.1	LLMs Applied to Project Management Processes	24
2.1.1	Bibliometric framing of the field	24
2.1.2	Process-level applications: planning, risk, and estimation	25
2.1.3	Cross-process benchmarks and adjacent contributions	26
2.2	LLMs Applied to Business Analysis and Requirements Engineering	28
2.2.1	Bibliometric framing of the field	28
2.2.2	Elicitation, extraction, and stakeholder requirements	28
2.2.3	Quality assurance, collaboration, and formalisation	29
2.3	LLM Benchmarking Methodologies	30
2.3.1	The foundational paradigms	30
2.3.2	Complex-system evaluation and hybrid automated-expert patterns	31
2.4	Research Gap Analysis	32
2.4.1	The absence of a unified cross-process evaluation framework	33
2.4.2	The heterogeneity and partial automatability of evaluation criteria	34
2.4.3	The domain-agnostic structure of foundational LLM benchmarks	34
2.4.4	Synthesis and positioning of the present work	35

3	Framework Design	36
3.1	Methodological Foundation	36
3.2	Process Selection: The Twelve Tasks	38
3.2.1	Empirical input from the first survey	38
3.2.2	Operational decomposition into twelve tasks	40
3.2.3	Bridge from the seven macro-processes to the twelve tasks	43
3.2.4	The twelve tasks	45
3.2.5	Limits of the present selection	46
3.2.6	Allocation of benchmark items across the twelve tasks	47
3.3	Operational Questions Design	47
3.3.1	Cognitive coverage of the operational questions	49
3.3.2	Illustrative benchmark items	49
3.4	Evaluation Criteria: Three Families of Methodological Trade-offs	57
3.4.1	Family A: granularity of correctness	57
3.4.2	Family B: compositional aggregation	58
3.4.3	Family C: error typology and asymmetric weighting	59
3.4.4	The full criteria table	59
3.4.5	Note on the underlying scoring techniques	60
3.5	The Closed Candidate List Design Pattern	60
3.6	Framework Deliverable: The Integrated Library	61
3.7	Operational Specification	62
4	Validation	68
4.1	Purpose and Strategy of the Validation	68
4.2	Method	69
4.2.1	Instruments	69
4.2.2	Respondents and Recruitment	69
4.2.3	Analytical Approach and Decision Rules	69
4.3	First Survey: Relevance of the Macro-Processes	70
4.3.1	Respondent Profile	70
4.3.2	Process Relevance	70
4.3.3	Relevance by Professional Role	71
4.3.4	Evaluation Criteria and Scoring Method	72

4.4	Second Survey: Validity of the Evaluation Criteria	73
4.4.1	Respondent Profile	74
4.4.2	Validation Results	74
4.4.3	Within-Process Comparison and Metric Selection	74
4.4.4	Consensus and Dispersion	75
4.4.5	The Single Decision Point	76
4.4.6	Robustness	77
4.4.7	Convergence of the Qualitative Evidence	78
4.5	Synthesis: The Validated Framework	78
5	Benchmark Execution and Results	80
5.1	Models Under Evaluation	80
5.2	Administration Protocol	81
5.3	Scoring Procedure	81
5.4	Aggregate Results	82
5.5	Per-Task Results	83
5.6	Task-Level Analysis	83
5.7	Integrity of the Comparison	86
5.8	Synthesis	86
6	Discussion	88
6.1	The Findings in Light of the Research Questions	88
6.2	Interpretation of the Benchmark Results	90
6.3	Implications	90
6.3.1	Methodological Implications	90
6.3.2	Implications for the Integrated Framework	91
6.3.3	Implications for Practice	91
6.4	Limitations	91
6.4.1	Face Validity Is Not Measurement Accuracy	91
6.4.2	Sampling	92
6.4.3	The Closed-Candidate-List Trade-off	92
6.4.4	Choice and Coverage of the Scoring Techniques and Criteria	92
6.4.5	The Scope of the Execution	92

6.5	Concluding Remarks	93
7	Conclusion	95
7.1	Summary of Contributions	95
7.2	Answer to the Research Questions	96
7.3	Toward Domain-Specific Evaluation of Generative Models	97
7.4	Future Work	97
	References	99
A	First Survey Instrument and Synthesis	103
B	Automated Scoring Pipeline	104
C	Second Survey Instrument	106
D	Benchmark Item-Bank	108

List of Figures

3.1	Allocation of benchmark items across the twelve tasks, in proportion to the relevance of the corresponding macro-processes in the first survey. The same axes carry the observed model performance reported in Chapter 5.	48
4.1	First survey — relevance of the seven macro-processes for model-supported evaluation, ordered by mean, with 95% confidence intervals. The dashed line marks the 4.0 core-relevance threshold.	72
4.2	First survey — mean relevance rating of each macro-process by professional role. The dashed line marks the 4.0 threshold.	73
4.3	Second survey — mean appropriateness rating of the twenty-four criteria, ordered by mean, with 95% confidence intervals. The dashed line marks the 4.0 threshold and the shaded band the 3.85–4.0 tolerance zone.	75
4.4	Second survey — the two candidate criteria for each task. The circled marker is the higher-rated criterion, adopted as the task’s official metric.	77
5.1	Mean official-metric score by discipline and overall, for the two models across the full instance pool of 998 instances.	83
5.2	Per-task official-metric scores for the two models across the twelve benchmark tasks. The axes are those of Figure 3.1; the radial coordinate is the mean official-metric score on the unit interval.	85

List of Tables

3.1	Bridge from the seven macro-processes evaluated in the first survey to the twelve operational tasks of the framework.	43
3.2	The twelve tasks of the framework, with taxonomy coordinates and derivation from the surveyed macro-processes.	64
3.3	The five output-structure classes of the framework.	65
3.4	Cognitive coverage of the twelve tasks, read through the levels of Bloom’s revised taxonomy.	66
3.5	The twenty-four evaluation criteria of the framework, classified by family of trade-off.	67
4.1	First survey — respondent profile (N = 40).	71
4.2	First survey — appropriateness of the evaluation criteria (mean rating, N = 40).	73
4.3	Second survey — respondent profile (N = 33).	74
4.4	Second survey — validation results for the twenty-four criteria (N = 33). Verdict per the decision rule of Section 4.2.3.	76
4.5	Validated scoring layer — the official metric adopted for each task.	79
5.1	Per-task benchmark results over the full instance pool. Score is the mean of the task’s official metric (Table 4.5) across the task’s instances.	84
6.1	Research questions and the evidence that answers them.	89
6.2	Principal limitations, their mitigation, and their disposition.	93
D.1	Composition of the benchmark item-bank across the three generation waves.	108

Chapter 1

Introduction

1.1 Background and Motivation

The rapid maturation of Large Language Models (LLMs) in the period 2022–2026 has produced an inflection in how organisations approach knowledge-intensive work. Tasks that until recently required hours of human deliberation, including the drafting of project documentation, the synthesis of stakeholder requirements, and the structured analysis of project risks, are now routinely attempted with generative AI assistance. The professional disciplines of Project Management (PM) and Business Analysis (BA) have entered this transition unevenly: a growing body of empirical and applied research demonstrates that LLMs can support specific PM and BA activities with non-trivial competence, while the methodological apparatus for assessing how well they support these activities has not kept pace.

The asymmetry between adoption and assessment is the empirical motivation of the present thesis. The literature reviewed in Chapter 2 documents thirty publications between 2021 and 2026 that apply LLMs to PM and BA tasks ranging from planning and scheduling to requirements elicitation, risk classification, and stakeholder analysis. Each contribution typically reports performance on a single task using a metric chosen by the authors, often without explicit grounding in either a canonical professional taxonomy or a reproducible scoring procedure. The result is a fragmented research landscape in which conclusions about LLM capability are difficult to compare across studies, and in which practitioners who would like to know which LLM performs best on, for example, work breakdown structure design or risk identification have no principled basis for the answer.

The methodological motivation of the thesis follows from the empirical one. The founda-

tional LLM benchmarking literature, exemplified by the Holistic Evaluation of Language Models framework (Liang et al., 2023), the Massive Multitask Language Understanding benchmark (Hendrycks et al., 2021), and the Beyond the Imitation Game collaborative effort (Srivastava et al., 2023), has established the multi-metric scenario-by-criterion paradigm as the methodological standard for LLM evaluation. None of these foundational benchmarks, however, contains scenarios specific to Project Management or Business Analysis, and the single PM-anchored benchmark identified in the literature (Karnouskos, 2024) adopts a multiple-choice format borrowed from the Project Management Professional certification examination, testing knowledge recognition rather than the operational capacity to produce PM and BA artefacts.

The combination of these two observations defines the methodological space in which the present thesis intervenes. A domain-specific framework for evaluating LLMs on PM and BA processes is needed because the methodological apparatus of generic benchmarks, though sound, has never been specialised to the operational tasks of these disciplines. The framework that this thesis proposes is therefore neither a competitor to the foundational benchmarks nor a replacement for the single PM-anchored benchmark in the literature; it is an instrument that inherits the methodological commitments of holistic evaluation, specialises them to the PM and BA domain through a structured taxonomy of twelve benchmark tasks and twenty-four fully automatable evaluation criteria, and is carried through to a concrete comparative execution against contemporary models.

The professional disciplines whose evaluation is at stake deserve a moment of framing. The Project Management Institute defines Project Management as the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements; its canonical reference, the seventh edition of the PMBOK Guide, organises Project Management work into Performance Domains (Stakeholders, Team, Development Approach, Planning, Project Work, Delivery, Measurement, and Uncertainty) crossed with the temporal Process Groups from Initiating to Closing (Project Management Institute, 2021). Business Analysis is organised, in turn, by the PMI Guide to Business Analysis into Knowledge Areas (Needs Assessment, Stakeholder Engagement, Elicitation, Analysis, Traceability and Monitoring, Solution Evaluation) crossed with a parallel set of Process Groups that includes a Defining and Aligning phase preceding Initiating (Project Management Institute, 2017). While the two disciplines overlap substantively in their treatment of stakeholders and requirements, they diverge in their core orientations: PM is concerned with the delivery of projects under constraints, whereas BA is concerned

with the structured analysis of business needs and requirements that projects must satisfy. The framework developed in this thesis covers six Performance Domains on the PM side and three Knowledge Areas on the BA side, with a deliberate balance between the two disciplines.

1.2 Research Problem

The literature reviewed in Chapter 2 surfaces three distinct but mutually reinforcing research gaps that together define the research problem of the present thesis. The gaps are stated here in compressed form and developed in full in Section 2.4.

The first gap is the absence of a unified cross-process evaluation framework: the existing literature offers bibliometric mappings and benchmarks each confined to a single task or a narrow cluster, but none integrates PM and BA processes within a shared, standards-grounded taxonomy, operationalizes them through reproducible structured outputs, and validates the design with practitioners of both communities.

The second gap concerns the metrics through which performance is measured: automated measures such as MAE and RMSE are established within the numerical estimation sub-domain, but outside it the field relies decisively on expert rubrics, inter-rater agreement, or qualitative coding, so that no unified set of fully automatable criteria spans the diversity of professional activities that practitioners actually perform.

The third gap concerns the position of PM and BA work relative to the foundational LLM evaluation literature. The multi-metric scenario-by-criterion paradigm established by Liang et al. (2023) and the task-diversity principle established by Srivastava et al. (2023) supply the methodological apparatus required to address the first two gaps, but none of the foundational benchmarks contains scenarios specific to PM or BA. The methodological tools therefore exist; their adaptation to the domain has not yet been carried out.

The research problem of the present thesis is the joint resolution of these three gaps through the construction, operationalization, empirical validation, and execution of a methodological framework that integrates twelve PM and BA benchmark tasks within a single taxonomy, operationalizes each task through a closed-input structured-output specification, scores each output through fully automated criteria that require neither expert rubric application nor human evaluator coordination at scoring time, and applies the resulting instrument to a comparison of contemporary LLMs.

1.3 Research Questions

The research problem defined in Section 1.2 is decomposed into four formal research questions that organise the contribution of the thesis and structure its empirical validation and execution.

RQ1 — Process selection. Which Project Management and Business Analysis processes are perceived by practitioners as relevant and well-suited to LLM-based support, and how should such processes be operationalised as benchmark tasks compatible with fully automated evaluation? RQ1 is empirical at the practitioner-validation stage (which processes do practitioners endorse) and methodological at the operationalisation stage (how to translate practitioner-validated processes into benchmarkable units with structured inputs and outputs). Chapter 3 answers RQ1 through the two-stage selection procedure that transforms seven macro-processes empirically validated by the first survey into twelve operational tasks through operational decomposition and dual framing.

RQ2 — Evaluation criteria. For each benchmark task identified under RQ1, what are the candidate evaluation criteria that meet the joint requirement of full scoring automation and methodological validity from the perspective of PM and BA practitioners? RQ2 is methodologically prior to any cross-model comparison of LLM performance: a benchmark whose evaluation criteria are not professionally validated produces scores whose meaning is unclear, regardless of how rigorously the scoring procedures are implemented. RQ2 is addressed in Chapter 3 through the twenty-four evaluation criteria organised into three families of methodological trade-offs (granularity of correctness, compositional aggregation, error typology) and submitted to a second survey for practitioner validation.

RQ3 — Empirical validation. To what extent does practitioner consultation, structured as two surveys covering the relevance of processes and the methodological validity of evaluation criteria respectively, supply the empirical evidence required to support the framework as a defensible instrument for LLM benchmarking on PM and BA tasks? RQ3 is the meta-methodological question that closes the loop between framework design and empirical validation. Chapter 4 addresses RQ3 through the two-survey protocol and the analysis of the resulting practitioner judgements.

RQ4 — Comparative execution. When the validated framework is applied to contemporary general-purpose LLMs, what differences in task-level and domain-level performance does it reveal, and does the instrument produce a meaningful and reproducible spread of scores across models of differing capability? RQ4 moves the inquiry from the validity of the instrument to its discriminating power in use: a benchmark that has been professionally validated but never executed demonstrates methodological soundness without demonstrating that its criteria, applied to real model outputs, separate stronger performance from weaker performance in an interpretable manner. Chapter 5 addresses RQ4 by administering the framework to two contemporary LLMs over the full instance pool under a fixed protocol, scoring every output through the deterministic pipeline, and reporting the resulting comparison by task and by discipline.

The four research questions are non-overlapping but mutually dependent. RQ1 supplies the set of tasks on which RQ2 operates; RQ2 supplies the criteria on which RQ3 elicits validation; RQ3 supplies the empirical basis on which RQ1 and RQ2 can be claimed as professionally defensible; and RQ4 supplies the evidence that the validated instrument, once executed, behaves as an instrument should. The thesis answers each question in turn, with explicit reporting of where the answer is empirically grounded, where it follows from methodological reasoning, and where the two combine.

1.4 Scope and Boundaries

The thesis is methodologically circumscribed in two senses that require explicit statement at the outset, because conflating either with the broader enterprise of LLM benchmarking would misrepresent both what the thesis claims and what it does not claim.

The first circumscription concerns the scope of the framework relative to the full PMBOK and PMI Business Analysis taxonomies. Six of the eight Performance Domains of PMBOK 7e are within scope (Stakeholders, Planning, Schedule as a sub-domain of Planning, Scope as a sub-domain of Planning, Risk, Integration, and the Finance dimension of Planning), as are three of the six Knowledge Areas of the PMI Guide to Business Analysis (Stakeholder Engagement, Elicitation, and Analysis). Remaining Performance Domains (Team, Development Approach, Project Work, Delivery, Measurement, and Uncertainty as a distinct domain) and remaining Knowledge Areas (Needs Assessment, Traceability and Monitoring, Solution Evaluation) are scoped out of the present work on three converging grounds: the scope con-

straints of an MSc thesis, the absence of clear practitioner signal in the first survey for activities outside the included scope, and the structural difficulty of applying the closed-candidate-list pattern to activities whose canonical output is free-form narrative (as discussed in Section 3.6). Such coverage is therefore representative rather than exhaustive: the framework demonstrates methodological coherence on twelve tasks distributed across the most professionally salient subset of PM and BA activities, and indicates through its extensibility (Section 3.6) how the same pattern can be carried forward to further domains in future work.

The second circumscription concerns the type of contribution the thesis makes. The framework is positioned as a methodological contribution to LLM evaluation research, not as a contribution to the application of LLMs in PM or BA practice. The thesis does not propose new techniques for using LLMs to perform PM or BA work and does not advocate for the deployment of LLMs in professional settings. The comparative results reported in Chapter 5 are a demonstration of the framework in use, not a general ranking claim or an endorsement of any model for professional deployment: they establish that the instrument discriminates between models of differing capability and produces interpretable, reproducible scores, not that any particular model is fit for unsupervised use in practice. The thesis advances the circumscribed claim that, if and when an organisation or research group wishes to evaluate LLMs on PM and BA tasks, the framework developed here provides a defensible, reproducible, and practitioner-validated instrument for doing so, and it substantiates that claim by executing the instrument end to end.

1.5 Contributions

The thesis advances five contributions to the methodological literature on LLM benchmarking.

The first contribution is the integration of Project Management and Business Analysis processes within a single evaluation framework grounded in the canonical taxonomies of the Project Management Institute. As Chapter 2 documents, the existing literature contains no precedent for systematic cross-disciplinary integration: published benchmarks either focus on PM activities without addressing BA (the closest case being Karnouskos, 2024) or focus on requirements engineering activities without addressing PM (the systematic mappings of Zade-noori et al., 2025 and the empirical benchmarks they review). Chapter 3 maps twelve operational tasks onto the PMBOK 7e and the PMI Guide to Business Analysis matrices through the

two-stage selection procedure described in Section 3.2, providing the first unified evaluation taxonomy for the two disciplines.

A second contribution lies in the design and specification of twenty-four fully automatable evaluation criteria for the twelve benchmark tasks, organised into three families of methodological trade-offs. Chapter 2 establishes that fully automatable evaluation is mature within the numerical estimation sub-domain but gives way to expert-rubric assessment outside that sub-domain. Through the closed-candidate-list design pattern (Section 3.5), which transforms open-ended elicitation tasks into closed selection tasks scored by direct identifier-set comparison, automated scoring is extended to the full set of twelve tasks. The three trade-off families (granularity of correctness, compositional aggregation, error typology) are framework-level patterns rather than per-task particularities; their cross-task recurrence organises the evaluation logic in a manner that supports both the empirical validation of Chapter 4 and the extensibility of the design to further tasks in future work.

The third contribution is the design and execution of a two-survey empirical validation protocol that filters the framework on two distinct dimensions. The first survey assesses professional relevance: practitioners are asked which PM and BA macro-processes are relevant and well-suited to LLM-based support, and the results define the priority map from which the operational task list is derived. A subsequent survey assesses methodological validity of the evaluation criteria: practitioners are asked, for each task, whether the proposed evaluation criterion is a sensible measure of professional accuracy on that task. While the two instruments answer different questions and apply different filters, they share the methodological commitment of grounding the design in practitioner judgement rather than in the design intuitions of any individual researcher. The protocol is described in detail in Chapter 4.

A fourth contribution is the explicit adaptation of the multi-metric scenario-by-criterion paradigm of foundational LLM evaluation (Liang et al., 2023) to the PM and BA domain. The holistic evaluation paradigm is specialised by organising the twenty-four criteria into four families that mirror the methodological structure of HELM: operational metrics (latency, token consumption, cost), general quality criteria (drawn from question Q8 of the first survey: accuracy, logical coherence, adherence to context, absence of fabrication, completeness, decision usefulness, clarity of expression), BA-specific quality criteria (drawn from question Q9 of the first survey), and robustness criteria (drawn from open-text themes in the first survey: assumption explicitness, refusal-to-fabricate, confidence calibration, conflict surfacing). The

four-family architecture is presented in Chapter 3 and constitutes the methodological bridge between the foundational benchmarking literature and the domain-specific instrument that the thesis delivers.

The fifth contribution is the execution of the validated framework against contemporary LLMs. The instrument is administered to two widely used general-purpose conversational models, Claude Sonnet 4.6 and ChatGPT GPT-5.5 Thinking (Standard), each accessed through its publicly available consumer interface, over the full pool of nine hundred and ninety-eight benchmark instances under a single fixed protocol; every output is scored by the deterministic pipeline using the official metric validated for each task. The result is the first comparative benchmark of contemporary LLMs across an integrated set of twelve PM and BA tasks, reported as a per-task and per-discipline leaderboard with full coverage for both models. Chapter 5 presents this execution, and in doing so converts the framework from a validated proposal into an instrument whose discriminating behaviour has been observed in practice.

The five contributions are mutually reinforcing. The first establishes the cross-disciplinary scope; the second establishes the operational evaluation infrastructure; the third establishes the empirical grounding of the design; the fourth establishes the methodological lineage; and the fifth demonstrates the instrument in use, reporting the comparative evidence that its execution produces. Together they position the thesis as a methodological contribution whose validated instrument is carried through to a concrete demonstration, in the sense defined in Section 1.4.

1.6 Thesis Structure

The remainder of the thesis is organised into six chapters that develop the contributions stated in Section 1.5.

Chapter 2 reviews the body of research that grounds the methodological contribution of the thesis. The review proceeds in four sections covering the application of LLMs to Project Management processes, the parallel literature on LLMs applied to Business Analysis and Requirements Engineering, the foundational benchmarking literature, and the formal articulation of the three research gaps. The chapter establishes the position of the thesis with respect to a fragmented and rapidly evolving research landscape and supplies the citations on which the framework design of Chapter 3 draws.

Chapter 3 develops the framework that constitutes the principal contribution of the thesis.

The chapter is organised in sections covering the methodological commitments that underpin the design, the two-stage selection procedure that derives the twelve operational tasks from the seven macro-processes of the first survey, the principles of operational question design and the five output-structure classes, the three families of methodological trade-offs that characterise the twenty-four evaluation criteria, the closed-candidate-list pattern as the central methodological device, the integrated library that consolidates the artefacts produced, and the operational specification through which the framework is rendered executable. Chapter 3 is the most substantial chapter of the thesis and contains the technical detail required to make the construct operationally complete.

Chapter 4 reports the empirical validation of the framework through two surveys with practitioner respondents. The chapter describes the survey instruments, the recruitment of respondents, the demographic characterisation of the resulting samples, and the analysis of the responses. In the first survey, the seven macro-processes are filtered for professional relevance with forty practitioners spanning Project Manager, Business Analyst, Project Management Office, Product Manager, and Consultant roles. In the second survey, the twenty-four evaluation criteria are filtered for methodological validity with a second cohort of practitioners. Per-process and per-criterion results are reported in the chapter, alongside cross-tabulations by professional role and a synthesis of the open-text responses that inform the four-family criteria architecture introduced in Chapter 3.

Chapter 5 reports the execution of the validated framework. The chapter describes the choice of models, the administration protocol under which the full instance pool is presented to each model, the deterministic parsing and scoring of the returned outputs, and the resulting comparison. Results are reported per task and aggregated by discipline, and the chapter examines the patterns the execution surfaces, including the tasks on which the two models converge, the tasks on which they diverge most sharply, and the relationship between task structure and observed performance.

Chapter 6 discusses the implications of the validation and the execution for the framework and identifies the limits of the present work. The chapter examines patterns in the practitioner responses that the per-question analysis of Chapter 4 surfaces (notably the domain-loyalty pattern in process relevance and the cross-role differences in evaluation method preferences), interprets the comparative results of Chapter 5 in light of the research questions, and discusses what the framework can and cannot claim on the basis of the combined evidence.

Chapter 7 closes the thesis with a synthesis of the contributions, a summary of the answers to the four research questions, an outlook on the broader implications of the framework for the integration of LLM evaluation methodology into professional disciplines beyond PM and BA, and the directions in which the executed benchmark can be extended.

The thesis is accompanied by appendices reporting the first survey instrument and its synthesis (Appendix A), the automated scoring pipeline that operationalises the framework (Appendix B), the second survey instrument (Appendix C), and the benchmark item-bank produced across the three generation waves (Appendix D). The bibliography lists the publications cited in the manuscript, organised in APA seventh edition format.

Chapter 2

Literature Review

This chapter reviews the body of research that grounds the methodological contribution of the present thesis. The review proceeds in four sections. Section 2.1 surveys the application of Large Language Models (LLMs) to Project Management processes, organising the corpus by process area in alignment with the knowledge areas of the PMBOK Guide. Section 2.2 surveys the parallel literature on LLMs applied to Business Analysis and Requirements Engineering, organising the corpus by phase of the requirements engineering lifecycle. Section 2.3 steps back from domain applications to examine the methodological apparatus available for evaluating LLM outputs, drawing on the foundational benchmarking literature and on emerging hybrid evaluation patterns. Section 2.4 synthesises the three preceding reviews into a formal articulation of the research gaps that motivate the framework developed in Chapter 3. The four sections share an analytical commitment: the review is not a catalogue of contributions but a structured argument that positions the thesis with respect to a fragmented and rapidly evolving research landscape.

The corpus comprises thirty publications spanning the period 2021–2026, with a strong concentration in the last twenty-four months. Twenty-four publications were identified through searches on Scopus, IEEE Xplore, the ACM Digital Library, and Google Scholar; six additional foundational and benchmark-oriented publications were added through a methodologically motivated enrichment described in Section 2.3. The corpus excludes generic LLM application work where the connection to Project Management or Business Analysis processes is incidental rather than substantive.

2.1 LLMs Applied to Project Management Processes

Project Management has emerged as one of the application areas attracting increasing scholarly attention to the use of LLMs. The literature, however, is dispersed across knowledge areas and methodologically heterogeneous: empirical benchmarks coexist with conceptual proposals, applied case studies, and partial systematic reviews. The section is organised into three groups: the bibliometric framing of the field, the process-level applications, and the cross-process benchmarks together with adjacent methodological enablers.

2.1.1 Bibliometric framing of the field

Khalil et al. (2025) provide the most comprehensive bibliometric and qualitative review of artificial intelligence applications to Project Management published to date. Combining co-citation analysis with content analysis, the authors identify thematic clusters spanning planning, decision-making, and resource management, and report a growing presence of LLMs among the technologies underpinning recent advances. Their review also surfaces a set of explicit gaps: limited empirical evidence, scarce attention to the later phases of the project lifecycle, and unresolved issues of data quality, transparency, and practitioner adoption. The work supplies the bibliometric context for the present thesis but does not extend to framework construction or empirical validation.

A complementary and methodologically more rigorous mapping is offered by Radliński & Swacha (2025), who systematically catalogue thirty primary studies on the use of LLMs for early-stage software project estimation between 2022 and 2026. Their analysis documents that over four-fifths of the corpus was published within the most recent twenty-four months and that institutional fragmentation is high. More importantly for the present thesis, the authors identify three methodological problems in the field: inconsistency across studies in the choice of evaluation measures, the absence of a standard evaluation dataset, and a discrepancy in the reference baselines against which LLM-based approaches are compared. These three problems recur throughout the literature reviewed in this chapter and provide the empirical foundation for the gap analysis of Section 2.4.

2.1.2 Process-level applications: planning, risk, and estimation

Planning and scheduling form the most populated area in the corpus, reflecting both the prominence of these activities in professional practice and the natural fit between planning tasks and the sequential reasoning capabilities of LLMs. Li et al. (2026) propose A4PS, an agentic AI-assisted advanced planning and scheduling framework that combines a multi-agent workflow with knowledge augmentation, retrieval-augmented generation, and chain-of-thought prompting; the authors report improvements over base LLM configurations on task success rate, solution accuracy, and the quality and executability of generated code. The multi-agent paradigm recurs in Wang et al. (2025), whose MAEF framework assigns problem definition, population generation, evolutionary optimisation, and evaluation to dedicated agents and is reported to outperform classical heuristic and metaheuristic baselines on resource-constrained project scheduling instances. The recurrence of multi-agent design across these two works foreshadows a cross-domain pattern that re-emerges in the Requirements Engineering literature of Section 2.2.

Two further contributions in this area are direct precedents for the framework of Chapter 3. Sarim et al. (2025) propose a benchmark that compares five LLMs on the generation of task flows from project documentation, varying prompting strategies (including an ISO 21502-guided variant) and introducing a Hybrid Semantic Similarity Metric reported to exhibit lower variance and stronger correlation with human assessment than traditional similarity metrics; the work benchmarks multiple models on a structured output specification and explicitly couples automatic scoring with human validation, a hybrid pattern discussed further in Section 2.3.2. Liu et al. (2026) introduce CART, a traceable zero-shot planning framework with adaptive replanning for LLM agents operating under incomplete information, reported to outperform prompt-based and policy-control baselines; its emphasis on traceability and replanning is conceptually relevant to the scope and schedule processes the present framework evaluates. The most prominent comparative benchmark in the area is Shahriary et al. (2025), who assess LLMs against human Scrum Masters on Agile project planning using a standardised template and expert evaluation across operational feasibility, task clarity, sprint organisation, and risk management; they report that some LLMs surpass human Scrum Masters on a subset of dimensions while exhibiting substantial inter-model variability and clear limitations on dynamic adaptation. The mixed findings are revisited in Chapter 6.

Risk management comprises two complementary contributions that together encapsulate

a tension recurring throughout the corpus. Erfani & Khanjar (2025) compare classical NLP models with LLM configurations including GPT-4 on the classification of construction project risks from unstructured text, reporting that few-shot GPT-4 achieves competitive performance without training data and behaves more evenly on imbalanced risk categories than the classical baselines; the result supports balanced selection-accuracy measures (the family that information retrieval refers to as F1-style measures, combining precision and recall into a single score) on closed-output classification, a principle the present thesis adopts for risk identification. In contrast, Martin et al. (2025) evaluate ChatGPT on the broader and more cognitively demanding task of construction risk assessment through case studies, interviews, and qualitative error analysis, reporting that the tool improves the speed and breadth of risk awareness but lacks contextual depth relative to senior expert judgement. The contrast between the two studies encapsulates the methodological pattern of the corpus: automatable evaluation is mature on structured classification but gives way to expert-driven assessment as soon as the task admits open-ended outputs.

Cost and effort estimation is methodologically distinctive because it is the only sub-domain in which automatable evaluation is the published norm. Carpenter et al. (2024) fine-tune GPT-3.5 on the ISBSG dataset of software project records and evaluate cost and duration prediction against expert-estimate baselines using RMSE and MAE, establishing empirically that LLMs are competitive on cost estimation when fine-tuned on standard datasets. The methodological characterisation of the sub-field is supplied by the mapping of Radliński & Swacha (2025), which confirms that MAE is the dominant regression metric (reported in slightly more than half of the surveyed studies), with MdAE, MMRE, MSE, RMSE, NMAE, and Pred(X) as secondary choices, story points and person-hours as dominant units, and user stories as the most frequent input artefact. Together, the applied precedent and the methodological synthesis allow the present framework to adopt tolerance-band scoring on cost estimation with full traceability to established practice.

2.1.3 Cross-process benchmarks and adjacent contributions

A small but methodologically important group attempts to evaluate LLMs across multiple Project Management processes rather than on a single task. Barcaui & Monat (2023) provide a qualitative section-by-section comparison of a GPT-4-generated project plan with one produced by a human Project Manager across scope, schedule, cost, resources, quality, stake-

holder, communications, and risk dimensions, concluding that AI and human approaches are complementary and that prompt engineering plays a substantive role in output quality; the work is among the earliest cross-process evaluations but, by design, produces no automatable metrics. Karnouskos (2024) provides the only cross-knowledge-area LLM benchmark anchored to a normative source of the Project Management Institute, evaluating ChatGPT-3.5, ChatGPT-4, and Google Bard on two hundred and fifty-two questions drawn from PMP certification preparation, with all three models reported above 82% accuracy. The author acknowledges two structural limitations: the multiple-choice format reduces complex decision-making to answer recognition, and the wide public availability of PMP preparation materials raises a non-negligible risk of test contamination. Both limitations directly motivate the design choices of the present framework, namely structured outputs over closed candidate lists rather than multiple-choice answers. Wysocki & Ochodek (2026) leverage LLM-based data augmentation for the automatic classification of recurring software development tasks, mitigating class imbalance on industrial datasets and introducing a domain-adapted Monthly Classification Error metric; the work bridges the supervised classification setting of risk classification and the cross-process ambition of the two preceding studies, illustrating that automatable metrics scale to multi-class settings when the output space is properly structured.

Five further publications address topics adjacent to Project Management and are reviewed to delimit the scope of the chapter and to acknowledge transferable contributions. Gu et al. (2025) study LLM-empowered dynamic scheduling for hybrid flow shops through multi-agent deep reinforcement learning; the terminology of scheduling is shared with Project Management, but machine assignment in production lines is structurally distinct from project scheduling, so the work is a boundary marker rather than an application. M. Zhang et al. (2026) adapt a small open-source LLM to question answering on construction contracts derived from FIDIC standards; the work is closest to procurement and contract management but does not address the contract lifecycle directly, though its dataset-construction method for low-resource domains is transferable. Matera et al. (2026) propose a human-centred conversational platform in which an LLM generates SQL queries, visualisations, and explanations, evaluated through a user study involving thirty-two domain experts including IT Project Managers; it supports reporting and monitoring rather than executing PM processes. Chen et al. (2025) design PMTutor, a chatbot with personalised adaptive learning for Project Management training, reported to complement learning and increase engagement; the work falls within Project Management education rather

than execution and is referenced in Chapter 6 in the context of practitioner adoption.

2.2 LLMs Applied to Business Analysis and Requirements Engineering

The Business Analysis literature is more concentrated than the Project Management literature of Section 2.1, with the majority of contributions focused on requirements engineering activities defined in the PMI Guide to Business Analysis. The section is organised into three groups: the bibliometric framing of the field, the elicitation and stakeholder-requirements activities, and the quality-assurance, collaboration, and formalisation activities.

2.2.1 Bibliometric framing of the field

Zadenoori et al. (2025) provide the principal systematic literature review on LLMs for Requirements Engineering and the bibliometric foundation for this section. Reviewing a corpus of primary studies, the authors document that LLM-for-RE research has concentrated principally on requirements elicitation and validation, with evaluations frequently conducted in controlled environments and with limited deployment in real organisational workflows. The review confirms that the field is methodologically less mature than the broader software engineering literature on LLM applications and identifies evaluation heterogeneity as a recurrent concern, in parallel to the corresponding observation made by Radliński & Swacha (2025) for the Project Management estimation sub-field. The two reviews converge on a common methodological diagnosis from independent vantage points, strengthening the argument advanced in Section 2.4.

2.2.2 Elicitation, extraction, and stakeholder requirements

Almeida et al. (2025) present the principal applied benchmark on the conversion of elicitation interviews into software requirements. Using real stakeholder interviews from two systems, the authors compare LLM-generated requirements against a human-generated benchmark and report that ChatGPT-4 achieves the strongest performance among the evaluated models while both models exhibit limitations on ambiguity handling and on non-functional requirements. The result is one empirical motivation for the design choice adopted in Chapter 3 for the elicitation process, which scopes the task to explicit needs only in order to penalise over-

interpretation while preserving the cognitive structure of the activity. A complementary contribution is Saleem et al. (2026), who propose ReqNet, an LLM-driven pipeline for automated requirements extraction from heterogeneous and unstructured documents, intended for scenarios in which the input is an unordered corpus rather than a clean transcript; both ReqNet and the present framework rely on structured extraction from natural-language input but constrain the output space differently, ReqNet through learned representations and the framework through a closed-candidate-list design pattern that admits fully automated scoring.

Mircea et al. (2026) study LLM-assisted revision of stakeholder requirement statements through an empirical evaluation in which participants first express requirements without assistance and then assess LLM-generated revisions, reporting that the revisions are judged superior to the unaided baseline on clarity, perceived completeness, alignment, and unambiguity. The design supplies an informative precedent: the separation of stakeholder identification from stakeholder classification, adopted in Chapter 3, preserves the analytical distinction between recognising the presence of a stakeholder and faithfully characterising the intent that stakeholder expresses.

2.2.3 Quality assurance, collaboration, and formalisation

Ellsel & Stark (2025) present practical workflows in which an LLM-based tool reformulates requirements and assesses their quality against predefined criteria, reporting that GPT-4o aligns closely with human quality assessment and suggesting that requirements quality is emerging as a benchmarkable capability; the validation step, however, still anchors in expert-curated reference judgements rather than a fully automated procedure. Lubos et al. (2024) extend the theme by evaluating whether an LLM can analyse requirements against the criteria specified in the ISO/IEC/IEEE 29148 standard (International Organization for Standardization, 2018), explain its judgement, and suggest improvements. The work is methodologically significant because it operationalises requirements quality in terms of an established normative standard, supporting the construction of defensible automated quality metrics; the empirical findings, reported through a study involving software engineering practitioners, confirm the feasibility of LLM-based quality assurance as a structured task while highlighting that operational acceptance depends on the scope of the requirement under revision.

Sami et al. (2026) study a multi-agent LLM system supported by human input for the generation and prioritisation of requirements, evaluated through mixed-method case studies,

interviews, and questionnaires; the authors report that the system can assist in identifying goal-aligned requirements but that human intervention remains essential for organisational adoption. The result echoes the cross-domain multi-agent pattern observed in the Project Management literature of Section 2.1.2, and its recurrence across the two literatures reinforces the methodological significance of the pattern. Finally, Beg et al. (2025) provide a short survey on the use of LLMs to assist the formalisation of software requirements and their downstream verification: tasks that convert natural-language requirements into formal logical or mathematical specifications admit verifiable evaluation through entailment and consistency checking, the cleanest automatable setting in the corpus. The present thesis adopts a comparable principle of structured outputs that admit verifiable scoring but extends it from formal specification to the wider operational requirements engineering tasks that practitioners perform in industrial settings.

2.3 LLM Benchmarking Methodologies

The two preceding sections surveyed the application of LLMs to Project Management and Business Analysis processes. The present section steps back from domain applications to examine the methodological apparatus available for evaluating LLM outputs in general, drawing on the foundational benchmarking literature that established the canon of LLM evaluation and on emerging hybrid patterns that combine automated scoring with expert curation.

2.3.1 The foundational paradigms

Liang et al. (2023) introduce the Holistic Evaluation of Language Models (HELM) framework, the most methodologically influential infrastructure in the LLM-evaluation canon. HELM organises assessment as a top-down taxonomy of scenarios crossed with desiderata, standardises prompting and adaptation strategies for reproducibility, and scores models on seven complementary metrics: accuracy, calibration, robustness, fairness, bias, toxicity, and efficiency. The authors report that, prior to HELM, models had been tested on a shared subset of approximately 18% of the core scenarios that were retrospectively standardised; HELM raises this shared coverage to 96% across thirty models. The sixteen core scenarios cover question answering, summarisation, sentiment analysis, and other text-processing tasks, but none is specific to Project Management or Business Analysis. HELM nonetheless supplies the methodological foundation that the present thesis adapts to the PM and BA domain through its scenario-by-criterion

architecture, in which the families of general quality, BA-specific quality, and robustness criteria specialise the desiderata of holistic evaluation.

Hendrycks et al. (2021) introduce Massive Multitask Language Understanding (MMLU), the canonical multi-task knowledge-testing benchmark that has driven public LLM leaderboards since 2021. MMLU covers fifty-seven domains spanning STEM, humanities, social sciences, and professional fields through approximately fifteen thousand nine hundred multiple-choice questions, evaluated under zero-shot and few-shot protocols; Project Management does not appear as a domain and Business Analysis processes are absent. The benchmark has progressively saturated as frontier models have exceeded 90% accuracy, and its mono-metric multiple-choice format has come to be recognised as a limit of the paradigm. The present thesis adopts a domain-specific, output-structured format that resists saturation through structural difficulty rather than knowledge breadth, in deliberate departure from the MMLU template. Srivastava et al. (2023) introduce Beyond the Imitation Game (BIG-bench), the most ambitious collaborative benchmark in the canon, with two hundred and four problem variants contributed by hundreds of authors across more than one hundred institutions and deliberate heterogeneity of test formats. Its principal methodological lesson is that benchmark longevity depends on the variety of test instances rather than their depth: diverse problem design slows saturation and preserves discriminating power across model generations. The lesson motivates the design choice of the present thesis to cover twelve heterogeneous PM and BA processes rather than a single representative task. The most recent stage of this trajectory reinforces the same reading from the side of instrument design: Center for AI Safety et al. (2026) introduce Humanity’s Last Exam, a closed-ended benchmark of two thousand five hundred expert-written questions across more than one hundred subjects, constructed in explicit response to the saturation of the MMLU template. Its methodological premise is that the durable contribution of a benchmark resides in its questions, its candidate answers, and its scoring criteria, which retain their value while measured performance shifts with every model generation; the present framework adopts the same premise at the scale of a professional domain, locating its contribution in the validated instrument rather than in the scores of any particular administration.

2.3.2 Complex-system evaluation and hybrid automated-expert patterns

Shaikh (2026) surveys the principal frameworks for orchestrating LLM-based multi-agent systems and reviews their evaluation strategies, identifying open challenges that include dynamic

decomposition, persistent memory, communication efficiency, explainability and trust, and the absence of standard benchmarks. The survey is relevant on two grounds: it characterises at the methodological level the multi-agent design recurring in the application literature (Sections 2.1.2 and 2.2.2), and its identification of the absence of standard multi-agent benchmarks frames the methodological space in which the present thesis intervenes.

A distinctive evaluation pattern has emerged at the intersection of full automation and full expert judgement, in which automated metrics are computed against expert-curated references. Snell et al. (2025) provide an illustrative case from the medical domain, evaluating Llama 3 70B and GPT-4 against a human gold standard on the construction of a structured dataset from social media data; the models achieve approximately 87.4% accuracy with respect to the gold standard, with high inter-rater reliability on demographic attributes and lower reliability on interpretive content. The methodological contribution is the application of accuracy and Cohen's kappa to a structured output evaluation, with the gold standard supplied by human annotation. The same hybrid pattern appears in the Hybrid Semantic Similarity Metric of Sarim et al. (2025), computed automatically but validated through correlation with human assessment, and in the regression-evaluation tradition exemplified by Carpenter et al. (2024) and the mapping of Radliński & Swacha (2025), in which automated metrics presuppose an expert-curated ground truth such as the actual project cost. None of these patterns is a counterexample to fully automated evaluation: each still requires expert curation upstream of scoring. The framework of Chapter 3 closes the loop by curating the closed-candidate-list reference at design time, so that scoring at evaluation time is fully automatic without sacrificing the rigour of expert grounding.

2.4 Research Gap Analysis

The preceding sections surveyed the application of LLMs to Project Management and Business Analysis tasks, the methodological apparatus available for evaluating LLM outputs, and the empirical evidence accumulated between 2021 and 2026. The review reveals a fragmented research landscape in which the question of whether LLMs can support PM and BA processes effectively is addressed through isolated, task-specific studies rather than through a coherent and reproducible evaluation framework. The present section makes the fragmentation explicit by articulating three distinct, mutually reinforcing research gaps, each stated as a formal claim and defended against the surveyed evidence.

2.4.1 The absence of a unified cross-process evaluation framework

The first gap concerns the structure of the research field itself. Across the surveyed publications, no single work proposes an evaluation framework that simultaneously satisfies three conditions: integration of Project Management and Business Analysis processes within a shared taxonomy grounded in established professional standards; operational specification of evaluation tasks through reproducible inputs, outputs, and scoring procedures; and empirical validation of the framework with a practitioner sample.

The mapping and review studies illustrate the boundary between charting the field and constructing an operational instrument. Khalil et al. (2025) synthesise AI applications to Project Management descriptively rather than prescriptively; Zadenoori et al. (2025) perform the parallel exercise for requirements engineering but remain confined to the Business Analysis side of the disciplinary divide; and Radliński & Swacha (2025) map the narrower sub-field of LLMs for software project estimation, identifying inconsistent evaluation measures, the absence of a standard dataset, and discrepant baselines. The three reviews demonstrate that the literature is organised in fragments by knowledge area, by sub-task, or by single profession, but never integrated. The empirical studies reinforce this: each evaluates a single task or a narrow cluster, whether task-flow generation (Sarim et al., 2025), risk classification (Erfani & Khanjar, 2025), Agile planning (Shahriary et al., 2025), plan comparison (Barcaui & Monat, 2023), or requirements generation (Almeida et al., 2025). The work closest to a cross-process benchmark, Karnouskos (2024), spans all ten PMBOK knowledge areas and reports composite accuracy above 82%, but its multiple-choice format and exposure to test-contamination risk mean that it measures recognition of normative knowledge rather than the operational capability to produce structured PM artefacts such as work breakdown structures, risk registers, stakeholder analyses, or schedules. This first gap is the strongest and most consequential of the three: the framework introduced in this thesis fills it by mapping twelve processes from the PMBOK Guide and the PMI Guide to Business Analysis into a single taxonomy, operationalising each as a closed-input, structured-output task, and validating the design through two practitioner surveys spanning both communities.

2.4.2 The heterogeneity and partial automatability of evaluation criteria

The second gap concerns the metrics through which LLM performance is measured. A preliminary formulation advanced early in the present work held that existing criteria are predominantly subjective and not fully automatable; confrontation with the evidence requires a more nuanced articulation. Automated metrics are well established for a specific subset of tasks, notably numerical estimation, where regression measures such as MAE and RMSE (Carpenter et al., 2024; Radliński & Swacha, 2025) and classification measures such as F1-score, recall, and precision (Erfani & Khanjar, 2025; Wysocki & Ochodek, 2026) are standard. Outside this subset, the landscape shifts decisively toward expert-driven assessment: studies of LLM-generated plans, risk assessments, requirement revisions, and elicitation outputs uniformly anchor evaluation in expert rubrics, inter-rater agreement, or qualitative coding (Barcaui & Monat, 2023; Shahriary et al., 2025; Martin et al., 2025; Mircea et al., 2026; Ellsel & Stark, 2025; Lubos et al., 2024; Sami et al., 2026). The hybrid approaches that occupy the middle ground (Sarim et al., 2025; Snell et al., 2025; Almeida et al., 2025) confirm that fully automated evaluation is possible only when an expert-curated reference exists; they are not counterexamples to the gap but evidence of its precise contour. The conclusion is therefore not that all evaluation in the field is subjective, but that the field lacks a unified set of fully automatable measures spanning the diversity of PM and BA tasks. The present thesis addresses this shortfall through twenty-four criteria, two per process, all computable from a structured output and requiring no semantic matching, no expert rubric, and no human evaluator at scoring time.

2.4.3 The domain-agnostic structure of foundational LLM benchmarks

The third gap concerns the position of the present work relative to the foundational LLM evaluation literature. General-purpose benchmarks, most notably HELM (Liang et al., 2023), MMLU (Hendrycks et al., 2021), and BIG-bench (Srivastava et al., 2023), establish the methodological apparatus of LLM evaluation but contain no scenarios or tasks specific to Project Management or Business Analysis. Their lessons remain directly relevant: the multi-metric paradigm of Liang et al. is the conceptual foundation for the four-family criteria architecture of the present framework; the task-diversity principle of Srivastava et al. motivates a design spanning twelve processes rather than a single task; and the contrast with the multiple-choice knowledge-testing format of Hendrycks et al. is precisely what the framework departs from in order to preserve

the cognitive structure of the processes it benchmarks. The single PM-specific benchmark in the corpus, Karnouskos (2024), inherits the multiple-choice paradigm from the PMP exam format and therefore demonstrates theoretical PMBOK knowledge without testing the capacity to produce structured artefacts. The third gap is thus not the absence of a benchmark covering PMBOK content but the absence of one that adapts the multi-metric, multi-scenario paradigm of holistic evaluation to the operational tasks that define PM and BA work in practice.

2.4.4 Synthesis and positioning of the present work

The three gaps jointly articulate the contribution of this thesis. The first establishes the absence of a unified cross-process framework; the second establishes that, where evaluation is conducted, the criteria are either restricted to sub-domains where automated metrics are mature or rely on expert judgement at scoring time; the third establishes that the foundational evaluation literature supplies the methodological apparatus required to address the first two but has not adapted it to the PM and BA domain.

The framework developed in this thesis responds to each gap directly and traceably. To address the first, it integrates twelve processes from the PMBOK Guide and the PMI Guide to Business Analysis into a single taxonomy validated through two practitioner surveys. To address the second, each process is operationalised as a benchmark task with a closed candidate list, a structured output, and two fully automatable criteria, twenty-four in total, designed to require neither semantic matching nor expert rubric nor human evaluator at scoring time. To address the third, the criteria are organised into four families that specialise the multi-metric paradigm of holistic evaluation to the domain: operational metrics, general quality, BA-specific quality, and robustness. The thesis therefore inherits the methodological commitments of the foundational benchmarking literature while extending them to a domain in which they have not yet been instantiated.

The integrated argument is that LLM evaluation on PM and BA processes requires a domain-specific approach precisely because the methodological apparatus of generic benchmarks, while sound, has not been adapted to the operational decisions that define professional practice in these fields. The present thesis designs, operationalises, and validates the instrument that makes a principled benchmark possible, and it carries that instrument through to a concrete comparative execution against contemporary models, reported in Chapter 5. The chapter that follows develops the framework in detail.

Chapter 3

Framework Design

This chapter presents the methodological framework that constitutes the principal contribution of the present thesis. The construct specifies a structured taxonomy of twelve Project Management and Business Analysis benchmark tasks derived from the seven macro-processes evaluated in the first practitioner survey, a library of operational questions intended to elicit LLM outputs on those tasks, and a system of twenty-four evaluation criteria, two per task, that score the outputs in a fully automated manner without recourse to expert rubrics or human evaluators at scoring time. Section 3.1 states the methodological commitments that underpin the architecture and defines the taxonomic frame derived from the canonical PMI sources. Section 3.2 describes the two-stage selection procedure that transforms the seven surveyed macro-processes into the twelve operational tasks and documents the rationale for each. Section 3.3 sets out the principles applied to the operational questions, the five output-structure classes, and the cognitive coverage of the resulting items. Section 3.4 introduces the three families of methodological trade-offs that characterise the evaluation criteria, illustrates each family with worked examples, and provides the full criteria table. Section 3.5 presents the closed-candidate-list pattern as the central methodological device that enables full scoring automation. Section 3.6 summarises the deliverable as an integrated library, and Section 3.7 renders that library executable and links it to the execution reported in Chapter 5.

3.1 Methodological Foundation

The framework rests on four methodological commitments, stated here so that the design choices documented in the remainder of the chapter can be evaluated against them. The first

is taxonomic grounding: every task in the framework is positioned within the canonical taxonomies of the Project Management Institute, the PMBOK Guide (7th edition) for Project Management processes (Project Management Institute, 2021) and the PMI Guide to Business Analysis for BA processes (Project Management Institute, 2017). The second is operationalisation through structured outputs: every task is associated with a concrete question whose expected output has a specified structure that admits automated comparison against a ground truth. The third is full automation of scoring: every evaluation criterion is computable by deterministic procedures over the LLM output and a pre-curated reference, without semantic matching, expert rubric application, or inter-rater coordination at the time of scoring. The fourth is empirical validation: the framework is not asserted as defensible by design alone but submitted to two empirical surveys with practitioners, the first survey filtering processes for professional relevance and the second filtering criteria for methodological validity.

The four commitments stand in deliberate contrast to the dominant evaluation patterns identified in Chapter 2. Taxonomic grounding distinguishes the framework from the bibliometric and mapping efforts surveyed in Section 2.1, which catalogue the field without anchoring evaluation tasks to a normative source. Operationalisation through structured outputs distinguishes the framework from the multiple-choice paradigm exemplified by Karnouskos (2024), for the reasons set out in Section 2.4.3. Full automation of scoring distinguishes the framework from the expert-rubric pattern that dominates the literature outside the numerical estimation sub-domain, as surveyed in Section 2.4.2. Empirical validation distinguishes the framework from purely design-based proposals, supplying the methodological evidence that the design choices respond to practitioner-grounded needs.

The PMBOK Guide (7th edition) organises Project Management through Performance Domains and Process Groups, with Performance Domains capturing thematic areas of project work (Stakeholders, Team, Development Approach, Planning, Project Work, Delivery, Measurement, and Uncertainty) and Process Groups capturing the temporal phasing from Initiating to Closing (Project Management Institute, 2021). Within the Planning Performance Domain, the seventh edition explicitly groups scope, schedule, and cost management as the three core planning sub-activities, treating cost estimation as one of the canonical planning artefacts that a project plan must produce. The PMI Guide to Business Analysis adopts a parallel architecture, with Knowledge Areas in place of Performance Domains and a slightly different sequence of Process Groups including a Defining and Aligning phase that precedes Initiating (Project Man-

agement Institute, 2017). The framework selects six Performance Domains and three Knowledge Areas as the scope of the present work, leaving outside scope further domains such as Communications, Procurement, and Solution Evaluation. This bounded scope is methodologically necessary: a framework that attempts to cover all PMI domains in a single thesis would dilute the depth of the per-task design, while a framework anchored to a thoughtfully selected subset can demonstrate methodological coherence on representative cases and indicate how the design can be carried forward to further domains in future work.

The mapping between the framework's tasks and the canonical taxonomies is presented in two views. The Project Management side is positioned on a Performance Domain $\times$ Process Group grid, the standard PMBOK 7e representation. The Business Analysis side is positioned on a Knowledge Area $\times$ Process Group grid, the standard PMI Guide to Business Analysis representation. Both grids serve as the canonical reference for the task identifiers used throughout the remainder of the chapter and are consolidated in the bridge of Table 3.1.

3.2 Process Selection: The Twelve Tasks

The framework comprises twelve benchmark tasks selected through a two-stage methodological procedure. The first stage takes the seven PM and BA macro-processes empirically validated by the first practitioner survey of the thesis as the complete input to the framework. The second stage transforms the seven macro-processes into twelve operational tasks through two principles: operational decomposition of multi-component macro-processes into independent benchmarkable sub-tasks, and dual framing of cross-discipline processes that appear in both the PMBOK Guide and the PMI Guide to Business Analysis with substantively different operational scopes.

3.2.1 Empirical input from the first survey

The first practitioner survey of the thesis, reported in detail in Chapter 4, was administered to forty respondents spanning Project Manager (N=14), Business Analyst (N=13), Project Management Office (N=7), Product Manager (N=3), and Consultant (N=2) roles. The survey instrument sampled seven candidate macro-processes drawn from the PMBOK and PMI Business Analysis taxonomies (Planning, Scheduling, Risk Management, Identify Stakeholders, Analyze Stakeholders, Elicit Needs, Baseline Requirements) and asked respondents to rate the

relevance of each to LLM benchmarking on a five-point Likert scale.

The empirical results, reported in full in Chapter 4 and summarised here, position the seven macro-processes in three bands. Four macro-processes received a mean relevance score above 4.0: Identify Stakeholders (mean = 4.30, SD = 0.65), Risk Management (mean = 4.18, SD = 0.87), Planning (mean = 4.00, SD = 0.96), and Baseline Requirements (mean = 4.00, SD = 0.88). Two macro-processes occupied the band between 3.5 and 4.0: Elicit Needs (mean = 3.93, SD = 1.02) and Analyze Stakeholders (mean = 3.80, SD = 0.72). One macro-process fell below the 3.5 threshold: Scheduling (mean = 3.28, SD = 1.04).

All seven macro-processes were retained for the framework. Six retentions are immediate: a mean above 3.5 on a five-point Likert scale with forty practitioners constitutes a substantive positive signal of professional relevance, and excluding macro-processes with such ratings would discard empirical evidence rather than respect it. The seventh retention, Scheduling, requires explicit justification because of its lower relevance score and because the task formulation quality assessed by Q6 placed Scheduling at the lowest position of the seven macro-processes (Q6 mean 3.10, SD = 0.98). Two considerations support its inclusion. First, the cross-tabulation by professional role (reported in Chapter 4) shows that Project Managers and Project Management Offices rate Scheduling appreciably higher (mean 3.71 for PM, 4.00 for PMO) than Business Analysts (mean 2.31 for BA), and the lower overall mean is largely driven by the BA segment of the sample whose practice does not directly involve schedule construction. Excluding Scheduling on the basis of an aggregate mean would discard a process that the practitioners who actually own it endorse with above-threshold ratings. Second, Scheduling is the macro-process with the strongest benchmarkability profile: a closed input (task list with durations and dependencies), a structured output (table of start and end days), and an objective ground truth (the optimal feasible schedule under the given constraints). The low Q6 formulation rating informs the design rather than the retention: the operational task developed for Scheduling in Section 3.2.2 redesigns the input specification with explicit durations and dependencies, directly addressing the weakness that the first survey flagged. The combination of role-specific high relevance, structural benchmarkability, and addressable formulation weakness supports retention.

A second empirical pattern documented by the first survey informs the second stage of the selection. The cross-tabulation of Q4 relevance scores by professional role reveals a domain-loyalty pattern in which each community rates the macro-processes it owns most highly, with

the by-role figures reported in Section 4.3.3. The pattern is methodologically informative because it indicates that PM and BA practitioners operationally frame shared macro-processes (notably Identify Stakeholders) through different domain-specific lenses, which motivates the dual-framing principle applied in the second stage.

3.2.2 Operational decomposition into twelve tasks

The seven empirically validated macro-processes are transformed into twelve operational tasks through two methodological principles, applied process by process.

Principle (a): operational decomposition of multi-component macro-processes. When a macro-process encompasses cognitively distinct sub-activities with different input/output structures, it is decomposed into separate benchmarkable units. The decomposition is justified on three converging grounds: the sub-activities require distinct cognitive skills (recognising critical milestones is different from forecasting a numeric cost figure); the sub-activities produce structurally different outputs (selection from a list versus a real number versus a chronological ordering), and a single composite metric would obscure performance differences across these distinct skills; and the decomposition aligns with the granularity of PMBOK and PMI Business Analysis sub-processes, which already enumerate the constituent activities under each performance area.

Three of the seven macro-processes undergo decomposition. The Planning macro-process, the largest in scope, is decomposed into three sub-tasks corresponding to the canonical planning artefacts that the PMBOK 7e Planning Performance Domain prescribes. Scope definition becomes the task Plan Scope Management / Define Scope (internal identifier PM-04), which evaluates the LLM's choice of work breakdown structure approach for a given project. Milestone identification becomes the task Develop Project Management Plan (internal identifier PM-05), which evaluates the LLM's selection and chronological ordering of critical milestones from a closed candidate roster. Cost estimation becomes the task Estimate Costs (internal identifier PM-EXT-01), which evaluates the LLM's quantitative prediction of project cost given a descriptive input document. All three sub-tasks are core components of the Planning Performance Domain in PMBOK 7e and are explicitly enumerated by the standard among the planning sub-activities; their separation reflects the fact that they engage substantively different cognitive skills (structural reasoning for scope, temporal sequencing for milestones, numerical

estimation for cost) and produce structurally different outputs (single choice, ordered selection, real number).

The Scheduling macro-process is decomposed into two sub-tasks reflecting the canonical predecessor-successor structure of the PMBOK 7e Schedule Performance Domain. Activity sequencing becomes the task Sequence Activities (internal identifier PM-EXT-02), which evaluates the LLM's identification of dependency relationships between project activities, with output structure a directed graph of (predecessor, successor) pairs. Schedule generation becomes the task Develop Schedule (internal identifier PM-03), which evaluates the LLM's assignment of start and end dates to tasks under duration and dependency constraints, with output structure a table of start and end days per task. The decomposition reflects the canonical PMBOK sequence in which sequencing precedes scheduling, and it allows independent evaluation of relational reasoning (dependency identification) and constraint-satisfaction reasoning (schedule construction).

The Baseline Requirements macro-process is decomposed into two sub-tasks reflecting the PMI Business Analysis distinction between classification and prioritisation. Requirements classification becomes the task Define and Elaborate Requirements (internal identifier BA-04), which evaluates the LLM's classification of pre-given requirements on two dimensions (functional versus non-functional, and MUST_HAVE / SHOULD_HAVE / NICE_TO_HAVE priority). Requirements prioritisation becomes the task Prioritize Requirements (internal identifier BA-EXT-01), which evaluates the LLM's ordering of pre-given requirements into a full ranked list. The two sub-tasks engage substantively different cognitive skills (categorical judgement on independent items versus relative ordering across a set) and produce structurally different outputs (classification labels versus a ranked sequence), justifying their separation as independent benchmark units. The Baseline Requirements macro-process is also retained on the PM side as PM-04 Plan Scope Management / Define Scope (already discussed under the Planning decomposition), since the survey item "Baseline Requirements" refers to a cross-discipline activity that PM practitioners frame as scope management and BA practitioners frame as requirements definition.

Principle (b): dual framing of cross-discipline processes. When the same conceptual process appears in both the PMBOK Guide and the PMI Guide to Business Analysis under different Knowledge Areas with substantively different operational framings, both framings are

retained as distinct benchmark units. The principle applies most directly to Identify Stakeholders, which appears in PMBOK 7e as a Stakeholders Performance Domain activity and in the PMI Guide to Business Analysis as a Stakeholder Engagement Knowledge Area activity. The two framings differ substantively. The PMBOK framing emphasises influence (HIGH-/MEDIUM/LOW) and internal/external positioning relative to the project. The PMI Business Analysis framing emphasises information role (PROVIDER/CONSUMER/APPROVER) and communication frequency requirements. The first survey's domain-loyalty pattern documented in Section 3.2.1 provides empirical support for the dual framing: PM and BA practitioners operationally frame stakeholders through different lenses, and a single benchmark unit applying only one framing would systematically misalign with one of the two communities. The Identify Stakeholders macro-process therefore yields two distinct tasks in the framework, PM-01 (PM framing) and BA-01 (BA framing), each tested with its own evaluation criteria.

The Analyze Stakeholders and Elicit Needs items of the first survey do not undergo decomposition. Both are scoped to operational units that the PMI Business Analysis taxonomy treats as single processes (Conduct Stakeholder Analysis under Stakeholder Engagement, Conduct Elicitation under Elicitation), and the empirical motivation for further decomposition is absent: the input artifact for each is a single document, the output structure is a single artifact (a two-by-two grid positioning or a selection from a closed list), and no methodologically distinct sub-skill is left unaddressed by treating the activity as one benchmarkable unit. Analyze Stakeholders becomes the task Conduct Stakeholder Analysis (internal identifier BA-02), and Elicit Needs becomes the task Conduct Elicitation (internal identifier BA-03). The Conduct Elicitation question is scope-narrowed to the selection of explicit needs from a closed candidate list, a design choice that resolves the tension between the elicitation activity (whose canonical output is a free-form narrative document, not directly benchmarkable through structured comparison) and the framework's commitment to fully automated scoring. The scope-narrowing is empirically supported by the BA-specific criteria results of question Q9 in the first survey, which rated the criterion "identification of explicit needs" at mean 3.90 and the criterion "uncovering of plausible implicit needs" at mean 3.75, the lowest of the seven BA-specific criteria assessed by the survey; the differential indicates that practitioners endorse the explicit-need restriction more strongly than the broader implicit-need uncovering, providing the empirical grounding for the design choice. Risk Management likewise enters the framework as a single benchmark unit, Identify Risks (internal identifier PM-02), since the input is a single project description,

the output is a ranked selection of risks from a closed candidate list, and the activity maps onto a single PMBOK Risk Performance Domain process at the granularity of benchmark evaluation.

3.2.3 Bridge from the seven macro-processes to the twelve tasks

Table 3.1 consolidates the two-stage selection procedure into a single synoptic view. Each row corresponds to one of the seven macro-processes evaluated in the first survey, reports its Q4 relevance mean, summarises the decision adopted in the framework, and gives the rationale of the operational decomposition or retention. The table makes the 1:1 derivation explicit at the level of the macro-process and is the canonical reference for the relationship between the empirical input of the first survey and the operational task list of the framework.

Table 3.1: Bridge from the seven macro-processes evaluated in the first survey to the twelve operational tasks of the framework.

Macro-process	Q4 mean	Operational outcome in framework	Rationale
Identify Stakeholders	4.30	PM-01 (PM framing) and BA-01 (BA framing)	Highest Q4 mean; dual framing motivated by the domain-loyalty pattern in the cross-tabulation by role (PMs and BAs operationally frame stakeholders differently).
Risk Management	4.18	PM-02 Identify Risks	Above-threshold Q4 mean and only macro-process classified KEEP AS-IS in the relevance-by-task-quality matrix of the first survey. Operationalised through a closed candidate list of twenty risks.

continued on next page

Table 3.1 (continued)

Macro-process	Q4 mean	Operational outcome in framework	Rationale
Planning	4.00	Three operational tasks: PM-04 (Plan Scope Management), PM-05 (Develop Project Management Plan, milestone identification), Estimate Costs (quantitative cost prediction)	Above-threshold Q4 mean. The PMBOK 7e Planning Performance Domain treats scope, schedule, and cost as the three core planning sub-activities; the operational decomposition reflects this taxonomic structure. The three sub-tasks are cognitively distinct (structural reasoning, temporal sequencing, numerical estimation) and produce structurally different outputs (single choice, ordered selection, real number).
Baseline Requirements	4.00	Two operational tasks: BA-04 (Define and Elaborate Requirements, classification) and Prioritize Requirements (full ranking)	Above-threshold Q4 mean and high BA-specific score (4.85 among BA respondents alone). The PMI Guide to Business Analysis Analysis Knowledge Area distinguishes classification (functional/non-functional and priority labels) from prioritisation (ranking), and the operational decomposition reflects this distinction.
Elicit Needs	3.93	BA-03 Conduct Elicitation, scope-narrowed to explicit-need selection	Below the 4.0 threshold but high BA-specific score (4.92 among BA respondents alone). Originally flagged as potentially non-benchmarkable because of the free-form narrative output of the canonical elicitation activity. The framework restricts the operational question to selection of explicit needs from a closed candidate list, enabling automated scoring; the restriction is empirically supported by the Q9 differential between explicit needs (3.90) and implicit needs (3.75).

continued on next page

Table 3.1 (continued)

Macro-process	Q4 mean	Operational outcome in framework	Rationale
Analyze Stakeholders	3.80	BA-02 Conduct Stakeholder Analysis	Below the 4.0 threshold but with strong BA-specific endorsement (4.38 among BA respondents). Operationalised as a closed two-by-two positioning task on the influence/interest grid.
Scheduling	3.28	Two operational tasks: PM-03 (Develop Schedule, generation) and Sequence Activities (dependency graph identification)	Below the 4.0 threshold on aggregate but appreciably higher in the PM and PMO segments of the sample (3.71 and 4.00 respectively). Retained for strong benchmarkability (closed structured outputs) and decomposed into two sub-tasks reflecting the canonical predecessor-successor structure of the PMBOK 7e Schedule Performance Domain. The low Q6 task formulation rating (3.10) is addressed in the operational task design by introducing explicit duration and dependency input specifications.

The net effect of the two-stage procedure is a framework with twelve operational tasks formally positioned in the PMBOK 7e and PMI Guide to Business Analysis taxonomies, each traceable to one of the seven macro-processes evaluated in the first survey through either a one-to-one mapping (Identify Risks, Conduct Stakeholder Analysis, Conduct Elicitation), a one-to-two operational decomposition (Scheduling, Baseline Requirements), a one-to-three operational decomposition (Planning), or the dual-framing principle (Identify Stakeholders, retained as both PM-01 and BA-01). The empirical and methodological warrants for each derivation are reported in Table 3.1; the resulting twelve-task list is summarized in the next subsection.

3.2.4 The twelve tasks

Table 3.2 summarizes the twelve tasks with their identifiers, types, taxonomy coordinates, expected output types, and the macro-process from which each derives. The table is the operational backbone of the framework: every subsequent section of this chapter references tasks by their identifier, and the empirical validation in Chapter 4 reports per-task results in the same identifier ordering.

The internal identifiers (PM-01 through BA-EXT-01) function as traceability codes only; the substantive identity of each task is given by the task name and the macro-process column of Table 3.2. The codes preserve

continuity with the framework's working documents and with the second-survey instrument that elicits practitioner judgements on the evaluation criteria for each task.

The resulting selection is taxonomically balanced. Seven tasks are positioned on the Project Management side and cover six Performance Domains of PMBOK 7e: Stakeholders (PM-01), Risk (PM-02), Schedule (PM-03 and Sequence Activities), Scope (PM-04), Integration (PM-05), and the Finance dimension of Planning (Estimate Costs). Five tasks are positioned on the Business Analysis side and cover three Knowledge Areas of the PMI Guide to Business Analysis: Stakeholder Engagement (BA-01 and BA-02), Elicitation (BA-03), and Analysis (BA-04 and Prioritize Requirements). The cross-PM/BA balance of seven-to-five reflects both the substantive emphasis of the first survey, which sampled four BA-side and three PM-side macro-processes, and the operational decomposition of the Planning macro-process into three sub-tasks (which sits entirely on the PM side). The framework is therefore not skewed in favour of either community, and the empirical validation of Chapter 4 can be reported with comparable resolution across the two domains.

3.2.5 Limits of the present selection

The twelve-task selection is not exhaustive of the PMBOK and PMI Business Analysis taxonomies, nor of the activities that practitioners might wish to see benchmarked. Future extensions of the framework, suggested by the open-text responses to question Q5 of the first survey and reported in Chapter 4, include three areas of practitioner interest. Requirements Validation and Verification, the BA-side activity of checking that elicited requirements satisfy quality criteria such as testability and consistency, was mentioned by several BA respondents but was scoped out of the present framework because its evaluation requires either a free-form quality assessment or a check against an external normative reference, both of which fall outside the closed-candidate-list pattern adopted in Section 3.5. Status Reporting and Executive Communication, the PM-side activity of producing concise project updates for senior stakeholders, was the most frequently mentioned theme in the open responses but produces free-form narrative output whose evaluation requires a scoring paradigm different from those developed here. Lessons Learned and Knowledge Management, an activity that spans both PM and BA practice, was mentioned by a smaller subset of respondents but suffers from the same free-form output structure.

Two reasons govern the deferral of these areas to future work. The first is the scope constraint of an MSc thesis, which would not allow the per-task design depth that the twelve included tasks receive if the selection were broadened further. The second is the closed-candidate-list pattern adopted as the central methodological device of the framework: free-form narrative output is fundamentally incompatible with the pattern's logic of direct identifier-set comparison, and benchmarking such output would require a scoring infrastructure (e.g., LLM-as-judge, expert rubric, or semantic similarity to a reference text) whose methodological soundness is itself unresolved in the current literature (see Section 2.3.2). Future work that addresses these activities will therefore need to extend the framework's scoring repertoire, not merely add new tasks under the existing one.

3.2.6 Allocation of benchmark items across the twelve tasks

The relevance evidence of the first survey does not stop at determining which tasks enter the framework; it also governs how much benchmark material each task receives. Rather than allocate an equal number of items to every task, the framework distributes its benchmark items across the twelve tasks in proportion to the relevance that the corresponding macro-processes obtained in the first survey, so that the resolution of the measurement is concentrated where professional relevance is highest. Figure 3.1 presents this allocation. The tasks derived from the most strongly endorsed macro-processes, risk identification, elicitation, and stakeholder analysis, carry the largest share of items, while the tasks derived from less strongly endorsed processes carry proportionally fewer; the shape of the profile is therefore a direct visual expression of the first survey’s relevance signal, carried forward into the design of the operational questions that the next section specifies.

The same axes provide the frame on which observed performance is read. The allocation established here fixes how much benchmark material each task carries; the per-task performance that the evaluated models obtain on these tasks is reported against this allocation in Chapter 5. The allocation profile depends only on the first survey’s relevance signal, whereas the observed-performance profile depends additionally on the model under evaluation and on the scoring criterion adopted for each task, and is therefore presented together with the execution rather than with the design.

3.3 Operational Questions Design

For each of the twelve tasks, the framework specifies one operational question that, together with an input artefact and a structured output specification, fully defines the benchmark task that an LLM is asked to perform. The design of the questions is governed by four principles, each of which addresses a methodological weakness identified in the literature reviewed in Chapter 2; one representative instance of each task is reproduced in Section 3.3.2.

The first principle is closed input specification. Every question is paired with an input artefact whose content is fully specified in advance: a project description document, a stakeholder roster, a list of candidate risks, or a transcript of an elicitation interview. The closure of the input eliminates the variability that would otherwise be introduced by free-form retrieval or by the LLM’s interpretation of an under-specified prompt, and ensures that performance differences across models are attributable to model capability rather than to prompt sensitivity. This commitment responds to the observation in Radliński & Swacha (2025) that the absence of a standard evaluation dataset is one of the three principal methodological problems in the LLM-for-estimation literature.

The second principle is structured output specification. Every question prescribes the exact format of the expected output: a single real number, a list of selected identifiers, a table with specified columns, an ordered ranking, or a set of ordered pairs encoding a dependency graph. The structural prescription transforms the scoring problem from one of natural-language comparison, where semantic equivalence is ill-defined, to one of structural comparison, where two outputs either match on a clearly specified relation or they do not. This commitment is the methodological device that enables full scoring automation, the central design goal of the framework.

The third principle is the existence of an ex-ante ground truth. Every question is paired with a canonical correct answer constructed in advance from the input artefact and the taxonomic conventions of the PMBOK Guide or

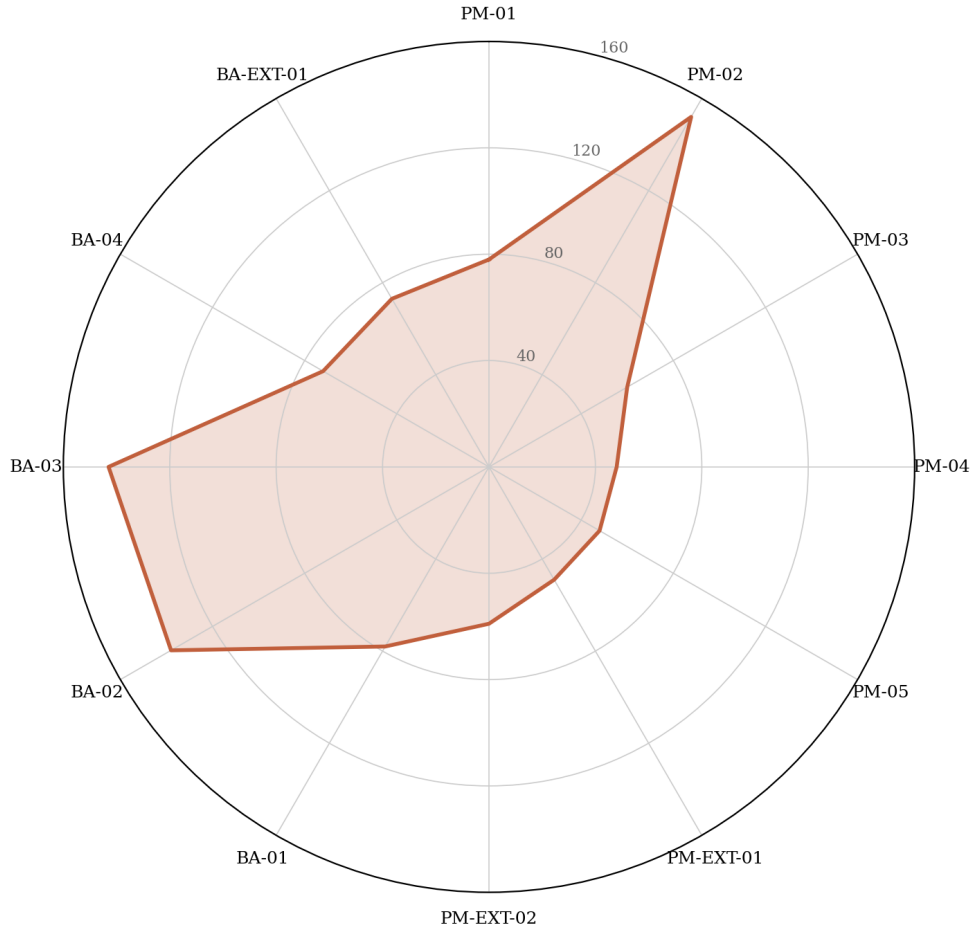


Figure 3.1: Allocation of benchmark items across the twelve tasks, in proportion to the relevance of the corresponding macro-processes in the first survey. The same axes carry the observed model performance reported in Chapter 5.

the PMI Guide to Business Analysis. The ground truth is the reference against which the LLM output is compared at scoring time, and its prior construction is a precondition for the deterministic scoring procedures of Section 3.4: a benchmark in which the correct answer is established post hoc, after seeing the LLM output, is methodologically vulnerable because the construction of the reference can be biased by the response under evaluation. The framework therefore commits to constructing the ground truth at instance design time, independently of any LLM run.

The fourth principle is closed candidate list. For seven of the twelve tasks, the question is framed as a selection from a pre-defined inventory of candidate entities (stakeholders, risks, milestones, needs, requirements). The inventory contains a mixture of correct items, plausible distractors, and where relevant implicit or inferred items that the LLM is instructed not to select. The closed format eliminates the semantic-matching ambiguity that pervades open-ended LLM evaluation: an LLM selecting the identifier S03 either matches the ground-truth identifier S03 or it does not, regardless of whether the underlying entity is described as “Chief Executive Officer”, “CEO”, or “the head of the company”. The closed-list pattern is examined in detail in Section 3.5 because it is the single most consequential design device in the framework.

The five output-structure classes that the framework distinguishes are summarised in Table 3.3. The classes

are mutually exclusive and exhaustive of the outputs produced by the twelve tasks, and each class admits a methodologically distinct family of scoring approaches. The deliberate inclusion of all five classes ensures that the framework’s evaluation logic is tested across the full range of LLM output structures relevant to PM and BA work.

Two tasks appear in two classes because their questions combine selection with attribute classification (PM-01 and BA-01) or with chronological ordering (PM-05). The double assignment reflects the genuinely composite nature of the task, not an ambiguity of design: each criterion proposed for these tasks engages explicitly with either the selection component, the classification component, or a weighted combination of both, as Section 3.4 documents.

3.3.1 Cognitive coverage of the operational questions

The operational questions can be read not only by the structure of their output but by the cognitive demand they place on the model, and this second reading addresses directly the limitation of certification-style assessment identified in Chapter 2. Using the levels of Bloom’s revised taxonomy (Anderson & Krathwohl, 2001) as a lens, the twelve tasks distribute across the upper bands of the hierarchy rather than clustering at the level of factual recall that multiple-choice examination items test. Three of the tasks are predominantly of the Apply level, in that they require the execution of an established procedure on given inputs: estimating a project cost, constructing a feasible schedule, and deriving the dependency relation among activities. Seven are predominantly of the Analyze level, in that they require distinguishing, classifying, or examining elements against criteria: identifying and characterising the relevant stakeholders on each of the two sides, discriminating the salient risks from distractors, identifying and ordering the critical milestones of a plan, positioning stakeholders on an influence-interest grid, selecting the explicitly stated needs from a transcript, and classifying requirements by type and priority. Two engage the Evaluate level, in that they require a judgement of relative merit: choosing the most appropriate decomposition approach for a project, and ranking requirements by priority against stated criteria. Table 3.4 records this mapping.

The mapping is a reading of cognitive demand rather than a rigid classification: a task such as Define and Elaborate Requirements engages judgment as well as analysis, and the boundary between adjacent levels is not sharp. Its purpose is not to grade the tasks by difficulty but to make explicit that the framework’s items engage the application, analysis, and evaluation bands of the hierarchy, which is precisely the cognitive territory that recognition-based benchmarks of the kind set out by Hendrycks et al. (2021) and Karnouskos (2024) leave untested. This coverage is one of the substantive respects in which the framework departs from the certification-style precedent.

3.3.2 Illustrative benchmark items

The four design principles and the five output-structure classes are best made concrete by exhibition. This subsection therefore reproduces one representative instance for each of the twelve tasks, drawn verbatim from the item bank documented in Appendix D. For the tasks whose candidate inventories are long, the inventory is abridged for space and the omission is marked; every box closes with the ex-ante ground truth against which the official metric of the task scores the model’s response. The instances shown are the opening instance of each task block;

the complete bank, with the full prompt input and ground truth of every instance, accompanies the thesis materials in machine-readable form.

The Identify Stakeholders task (PM-01) pairs a project charter with a roster of twenty candidate stakeholders that mixes genuinely relevant parties with plausible distractors, and requires both selection and attribute classification:

Project Charter: deployment of a new ERP module across the European business units. Scope includes financials and procurement; project sponsor sits on the executive board. Compliance is reviewed by an external regulator. The end-user community is concentrated in finance and operations.

Given the following list of 20 candidate stakeholders, select those who are GENUINELY relevant stakeholders of this specific project. For each selected stakeholder, indicate (a) influence level: HIGH, MEDIUM or LOW; (b) internal/external affiliation: INTERNAL or EXTERNAL. Return one line per SELECTED stakeholder in the format 'StakeholderID | Influence | Internal/External'.

CANDIDATES (abridged):

S02 - End-user representatives
S03 - Regulatory authority liaison
S05 - Executive sponsor
S07 - Internal sports-club coordinator
S11 - Stationery supplier
S13 - PMO director
[... 14 further candidates ...]

GROUND TRUTH: S02, S03, S05, S06, S10, S13, S14, S17 selected, each with its influence level and affiliation (e.g. S05 | HIGH | INTERNAL).

The Identify Risks task (PM-02) presents a project context and twenty candidate risks, of which ten are salient to the project and ten are distractors; the model must both select and prioritise:

PROJECT CONTEXT: a 12-month B2B SaaS platform development project, distributed engineering team across three EU cities.

From the following list of 20 candidate risks, select the 10 risks that are MOST relevant to this specific project, and rank them by priority where rank 1 is the most critical. Return 10 lines in the

format 'Rank | RiskID'.

CANDIDATE RISKS (abridged):

R06 - regulatory non-compliance with GDPR data processing
R09 - integration failures with customer legacy systems
R13 - office cleaning contract expiry
R19 - key technical staff turnover during critical phases
R20 - core feature scope creep due to shifting customer
requirements
[... 15 further candidates ...]

GROUND TRUTH ranking: R20, R19, R12, R09, R16, R18, R06, R08,
R03, R05.

The Develop Schedule task (PM-03) supplies a task list with durations and precedence constraints and requires a feasible schedule; the instance is reproduced in full:

You are given the following tasks with their durations (in days) and dependencies. Produce a schedule by assigning a start day and an end day to each task. Each task must respect its duration and must not start before all its predecessors finish. Use day 0 as project start.

TASKS:

T1 | duration 9 day(s) | predecessors: -
T2 | duration 9 day(s) | predecessors: -
T3 | duration 2 day(s) | predecessors: T1
T4 | duration 3 day(s) | predecessors: -
T5 | duration 7 day(s) | predecessors: -
T6 | duration 4 day(s) | predecessors: T5
T7 | duration 5 day(s) | predecessors: T6

GROUND TRUTH: earliest-start schedule with optimal makespan 16 days
(e.g. T1: 0-9, T3: 9-11, T5: 0-7, T6: 7-11, T7: 11-16).

The Plan Scope Management task (PM-04) is the single-choice task of the framework: it asks for a judgement of the most appropriate Work Breakdown Structure decomposition approach for a given project context:

Project context: a multi-product SaaS suite delivering several independent customer-facing applications.

Which of the following Work Breakdown Structure decomposition approaches is most appropriate?

- A) by Product (deliverable / component)
- B) by Phase (lifecycle phase)
- C) by Location
- D) by Resource / Department
- E) Hybrid

Respond with a single letter.

GROUND TRUTH: A (best-practice decomposition for this context);
E accepted as partially correct.

The Develop Project Management Plan task (PM-05) requires the selection of the critical milestones from a candidate list and their chronological ordering:

PROJECT CONTEXT: a multi-quarter B2B SaaS product launch.

From the following list of 15 candidate milestones, identify the FIVE most critical milestones for this project and order them chronologically (position 1 = first in time). Return five lines in the format 'Position | MilestoneID'.

CANDIDATE MILESTONES (abridged):

- M03 - General-availability launch
- M06 - Brand logo finalized
- M08 - Closed-beta launch with pilot customers
- M14 - Minimum viable product internal demo
- M15 - Product requirements baseline approved
- [... 10 further candidates ...]

GROUND TRUTH chronological order: M15, M14, M08, M03, M07.

The Estimate Costs task (PM-EXT-01) is the numeric task of the framework: a structured project brief must be converted into a single-number cost estimate, scored by relative error against the template-computed reference; the instance is reproduced in full:

PROJECT BRIEF - Digital transformation
Sector: digital transformation.
Duration: 3 months from project start.

Team composition: 11 FTE.

Technical complexity: low (low delivery complexity).

Technology profile: standard tooling.

Geographic scope: multi-site domestic.

Materials and equipment overhead: approximately EUR 150,000.

Reference average monthly cost per FTE in this sector: EUR 13,800.

TASK: provide a single-number cost estimate for the project,
in euro.

GROUND TRUTH: EUR 650,900.

The Sequence Activities task (PM-EXT-02) requires the recovery of the dependency relation among a set of activities, returned as a set of predecessor–successor pairs; the instance is reproduced in full:

PROJECT TYPE: software dev initiative.

You are given the following activities. Identify, for each activity, the predecessors that must finish before that activity can start. Return the dependency graph as the set of (predecessor, successor) pairs.

ACTIVITIES:

- C - Design wireframes and UI mockups
- D - Obtain design sign-off from stakeholders
- E - Set up development environment and CI
- F - Develop backend services
- G - Develop frontend application
- H - Integrate frontend and backend
- I - Run user acceptance testing
- J - Deploy to staging environment

GROUND TRUTH edges: (C,D), (D,E), (E,F), (E,G), (F,H), (G,H),
(H,I), (I,J).

The Identify Stakeholders (BA) task (BA-01) mirrors its PM counterpart on the Business Analysis side, with a business case as input and an information-role and communication-frequency schema as the classification dimensions:

BUSINESS CASE: Customer-data platform consolidation.

The organisation operates four legacy customer databases across marketing, sales, customer service, and finance. The proposed business analysis effort aims to consolidate these into a single customer-data platform with shared definitions, lineage, and access policies, supporting analytics, regulatory reporting, and customer operations across the four functions.

From the following list of 15 candidate stakeholders, select those who are GENUINELY relevant to the business-analysis activities of this initiative. For each selected stakeholder, classify

- (a) information role: PROVIDER, CONSUMER, or APPROVER;
- (b) communication frequency required: HIGH, MEDIUM, or LOW.

Return one line per SELECTED stakeholder in the format 'StakeholderID | InformationRole | CommunicationFrequency'.

CANDIDATES (abridged):

- S01 - Marketing data steward
- S02 - Catering vendor account manager
- S07 - Data-platform architect
- S08 - Data protection officer
- [... 11 further candidates ...]

GROUND TRUTH: S01, S03, S04, S05, S07, S08 selected, each with its role and frequency (e.g. S07 | APPROVER | HIGH).

The Conduct Stakeholder Analysis task (BA-02) requires the positioning of a given set of stakeholders on an influence-interest grid, with no selection component; the instance is reproduced in full:

PROJECT CONTEXT: a regulatory-driven IT compliance programme.

Position each of the following stakeholders on an influence/interest matrix by assigning two values per stakeholder:

- (a) Influence: HIGH or LOW
- (b) Interest: HIGH or LOW

Return one line per stakeholder in the format 'Stakeholder | Influence | Interest'.

STAKEHOLDERS:

- Legal counsel
- Internal audit lead

Head of IT
End users (operational staff)
Procurement manager
Vendor account manager

GROUND TRUTH: Head of IT HIGH/HIGH; Legal counsel and Internal audit lead HIGH/LOW; End users and Vendor account manager LOW/HIGH; Procurement manager LOW/LOW.

The Conduct Elicitation task (BA-03) pairs an interview transcript with a candidate-need inventory that mixes explicitly stated needs, needs merely implied by the transcript, and unrelated distractors; the scope-narrowing of Section 3.2.3 is visible in the instruction to select the explicit needs only. By design, the transcript embeds the wording of the explicit candidate needs verbatim, so that explicitness is a property of the text rather than a matter of interpretation:

INTERVIEW EXCERPT - topic: field-service mobile workflow.

Interviewer: Can you walk me through the main needs you see?

Field-operations supervisor: Sure. There are a few things we really need. We need dispatchers must see updated job statuses in real time. We need barcode scanning to identify equipment. We need support for attaching photos to work-order entries.

Interviewer: Anything else that's important?

Field-operations supervisor: We need complete audit log of who modified what and when. We need ability to capture work-order updates without network connectivity. We need user interface in Italian and English. Those are the main ones for me.

From the following list of 18 candidate needs, select ONLY those that are EXPLICITLY mentioned in the interview excerpt above. Do NOT select needs that you could merely infer from context -- return only the explicit ones. Return one line per selected need with its identifier only.

CANDIDATE NEEDS (abridged):

N04 - AI-based fraud detection
N05 - dispatchers must see updated job statuses in real time
N08 - ability to capture work-order updates without network connectivity
N13 - barcode scanning to identify equipment
[... 14 further candidates ...]

GROUND TRUTH: N05, N08, N12, N13, N14, N15 (the explicit needs only).

The Define and Elaborate Requirements task (BA-04) requires the classification of fifteen requirements on the two dimensions of type and priority:

BUSINESS CASE: a B2C digital platform serving European customers, with regulatory exposure (GDPR, PSD2), a 1m+ customer base, and a strategic objective to grow revenue while maintaining compliance.

For each of the following 15 requirements, classify them on two dimensions:

(a) FUNCTIONAL or NON_FUNCTIONAL

(b) MUST_HAVE, SHOULD_HAVE, or NICE_TO_HAVE

Return 15 lines in the format 'ReqID | type | priority'.

REQUIREMENTS (abridged):

Req05 - log all administrative actions for audit

Req07 - provide an in-app onboarding tutorial for new users

Req10 - support 10,000 concurrent users without service degradation

Req12 - retain transactional data for at least seven years

[... 11 further requirements ...]

GROUND TRUTH (excerpt): Req05 NON_FUNCTIONAL, MUST_HAVE;

Req07 FUNCTIONAL, NICE_TO_HAVE; Req10 NON_FUNCTIONAL, SHOULD_HAVE;

Req12 NON_FUNCTIONAL, MUST_HAVE.

The Prioritize Requirements task (BA-EXT-01) states four prioritisation criteria explicitly and requires a full ranking of ten requirements, each described through its standing on the four criteria:

BUSINESS CONTEXT. Prioritization criteria: business value, urgency (deadline pressure), downstream dependencies, and implementation risk (risk reduces priority). Equal weight on all four.

Given the following 10 requirements, rank them in order of priority where rank 1 is the highest priority. Return 10 lines in the format 'Rank | ReqID'.

REQUIREMENTS (abridged):

```
Req01 - deliver an audit log of all administrative actions;  
        moderate business value, regulatory deadline this quarter,  
        isolated, blocks nothing, moderate implementation risk  
Req03 - support OAuth2 client-credentials grant for  
        service-to-service; high business value, regulatory  
        deadline this quarter, blocks several downstream items,  
        low implementation risk  
[... 8 further requirements ...]
```

GROUND TRUTH ranking: Req03, Req05, Req02, Req04, Req07, Req01,
Req09, Req08, Req06, Req10.

Read together, the twelve instances exhibit the design principles of this section in operation: every input is closed and self-contained, every output is prescribed down to its line format, every ground truth exists before any model is queried, and, where the task admits it, the candidate inventory mixes correct items with calibrated distractors. The instances also make visible why the scoring of Section 3.4 can be fully deterministic: each response is a set, an ordering, a table, a classification, or a number, and each is compared with its reference by elementary arithmetic alone.

3.4 Evaluation Criteria: Three Families of Methodological Trade-offs

For each of the twelve tasks the framework specifies two evaluation criteria, producing a total of twenty-four candidate scoring rules. The two-criteria design is not an artifact of redundancy but a deliberate methodological device: each pair expresses a substantive trade-off between two defensible approaches to scoring the same task, and the empirical validation reported in Chapter 4 elicits practitioner judgements on which member of the pair is more methodologically appropriate. The two-criteria architecture also responds directly to the preference for multi-method evaluation expressed by a majority of respondents in the first survey, and to the observation in Liang et al. (2023) that single-metric evaluation conceals trade-offs that practitioners need to surface before deployment.

Across the twelve pairs of criteria, three recurring patterns of methodological trade-off emerge. The three patterns are mutually distinguishable and collectively cover all 24 criteria, providing a principled organization of the framework’s evaluation logic. The subsections that follow introduce each pattern with worked examples drawn from the criteria specification.

3.4.1 Family A: granularity of correctness

The first family of trade-offs concerns the granularity with which correctness is measured. One member of the pair scores correctness on a continuous proportional scale, where small deviations from the ground truth produce

small score reductions and large deviations produce proportionally large reductions. The other member scores correctness on a discrete-band scale, where deviations within a pre-specified tolerance band yield a constant score and deviations beyond the band produce a step decrease. The trade-off is between sensitivity to small differences (the continuous variant) and alignment with professional decision-thresholds (the discrete variant).

The clearest instance of Family A is the Estimate Costs task. The continuous criterion computes the relative absolute error and converts it to a score $s = \max(0, 1 - |y - y^*|/y^*)$, where y is the LLM estimate and y^* the true cost. The discrete criterion maps the same relative error onto four tolerance bands ($\pm 10\%$, $\pm 20\%$, $\pm 30\%$, beyond) with scores 1.0, 0.75, 0.50, and 0 respectively, reflecting the rough-order-of-magnitude, budget, and definitive estimate categories used in cost estimation practice. The same logical structure appears in the Plan Scope Management task (PM-04), where one criterion gives partial credit for partially-accepted WBS choices and the other reduces the scoring to a strict binary on best-practice choices; and in the Develop Schedule task (PM-03), where one criterion computes a composite score weighting feasibility (0.6) and makespan efficiency (0.4), and the other applies a binary feasibility gate followed by a continuous makespan score.

The methodological question that Family A poses is whether scoring should reward gradations of correctness in proportion to their magnitude, or whether scoring should map onto the discrete thresholds that practitioners actually use in their professional judgement. The first survey did not directly address this question, which is one of the items the second survey is designed to validate.

3.4.2 Family B: compositional aggregation

The second family concerns the compositional structure of the evaluation. One member of the pair treats the task as a single atomic operation and scores it through a single metric. The other member decomposes the task into two or more sub-operations and aggregates the sub-scores through an explicit weighted combination, with the weighting parameter encoding a substantive judgement about the relative importance of the components.

The clearest instance of Family B is the Identify Stakeholders task on the PM side (PM-01). The atomic criterion scores selection alone through a balanced selection-accuracy measure on stakeholder identifiers that penalises both missed stakeholders and wrong selections symmetrically. The decomposed criterion separates selection (60%) from classification of influence and internal/external attributes (40%), with the weighting parameter encoding the judgement that identifying the right stakeholders is the primary skill and classifying them correctly the secondary skill. The same logical structure appears in the BA framing of stakeholder identification (BA-01), where the decomposed criterion separates selection from information-role classification using the same 60/40 split; in the Develop Project Management Plan task (PM-05), where the decomposed criterion combines milestone selection (70%) with chronological ordering of correctly selected milestones (30%); and in the Prioritize Requirements task, where the decomposed criterion combines top-three set match (40%) with global rank-agreement (60%).

The methodological question that Family B poses is whether evaluation should treat composite professional tasks as wholes or should decompose them into their constituent sub-skills. The decomposition approach exposes sub-skill performance separately, providing more diagnostic information about LLM capabilities, but introduces a weighting parameter whose value is itself a methodological choice requiring justification. The atomic approach

avoids the weighting question by treating the task as indivisible, at the cost of obscuring sub-skill performance. The second survey is designed to elicit practitioner preferences on this trade-off task by task.

3.4.3 Family C: error typology and asymmetric weighting

The third family concerns the symmetric or asymmetric treatment of errors. One member of the pair applies a symmetric metric in which all errors of a given type contribute equally to the score reduction. The other member discriminates between error types or attribute types, applying differential weights to reflect substantive professional asymmetries.

The clearest instance of Family C is the Conduct Elicitation task (BA-03). The symmetric criterion applies a balanced selection-accuracy measure to the selection of explicit needs from the closed candidate list, treating all false positives equally. The asymmetric criterion distinguishes two types of false positive: selecting an implicit-but-inferable need (weight 0.5, treated as a soft error) and selecting an unrelated distractor (weight 1.0, treated as a hard error). The asymmetry encodes the substantive professional judgement that over-interpreting a document is a different kind of error from fabricating unrelated content. The same logical structure appears in the Sequence Activities task, where one criterion applies a symmetric balanced-accuracy measure to all dependency edges and the other applies double weight to edges lying on the critical path; in the Define and Elaborate Requirements task (BA-04), where one criterion gives equal weight to the functional/non-functional and priority dimensions and the other adopts a 30/70 priority-weighted aggregation; and in the Conduct Stakeholder Analysis task (BA-02), where the asymmetric criterion gives 70/30 influence-versus-interest weighting on the influence/interest grid.

The methodological question that Family C poses is whether evaluation should treat all errors as equally damaging or should reflect the substantive professional asymmetries between error types. The asymmetric approach captures domain-specific judgements that practitioners apply implicitly when judging LLM outputs (a critical-path scheduling error is more damaging than a non-critical-path one; an unrelated need selection is more damaging than a soft over-interpretation), but it introduces explicit weighting parameters whose values must be empirically validated. The symmetric approach defers to standard information-retrieval and classification metrics whose theoretical properties are well-understood, at the cost of obscuring the professional asymmetries. The second survey is designed to elicit practitioner preferences on this trade-off.

3.4.4 The full criteria table

Table 3.5 reports the full set of twenty-four criteria, each labelled with its primary family. Every task appears with two criteria, each of which is grounded in either a different family of trade-off (the most common case) or in two different specifications within the same family. The cross-task variety of trade-offs is itself a methodological feature of the framework: by spanning all three families across the twelve tasks, the design ensures that the empirical validation in Chapter 4 surfaces practitioner judgements on the full range of trade-offs that arise in the design of LLM-evaluation metrics for PM and BA tasks.

The classification of each criterion to a primary family reflects the dominant methodological motivation of the design choice. Some criteria could plausibly be assigned to two families: the PM-03 criterion pair, for ex-

ample, contrasts a compositional aggregation (Family B) against a granularity choice (Family A), and the table assigns each member to its own primary family. The conceptual cleanliness of the three-family taxonomy is unaffected by these boundary cases, because the empirical validation surfaces practitioner preferences on each pair independently of its family label.

3.4.5 Note on the underlying scoring techniques

Three established information-retrieval and statistical techniques underpin the scoring procedures referenced in this chapter; they are stated here for reproducibility, even though the prose of the framework refers to them through their operational descriptors rather than their formal names. The balanced selection-accuracy measure that appears in fifteen of the twenty-four criteria is the harmonic mean of precision (the proportion of selected items that are correct) and recall (the proportion of correct items that are selected), known in information retrieval as the F1-score (Van Rijsbergen, 1979); its property of penalising both wrong selections and missed selections symmetrically is the reason it is adopted as the framework’s set-comparison measure. The rank-agreement measure used in the Prioritize Requirements task is the Spearman rank correlation coefficient (Spearman, 1904), the standard non-parametric statistic for monotonic association between two orderings, mapped onto the unit interval for use as a score. The top-positions-weighted ranking measure used in PM-02 is the normalised Discounted Cumulative Gain (nDCG), the standard information-retrieval metric for ranked lists in which errors at the top of the ranking are penalised more heavily than errors at the bottom through a logarithmic discount factor (Järvelin & Kekäläinen, 2002). The framework’s prose refers to these techniques by their operational properties (balanced selection accuracy, rank-agreement measure, top-positions-weighted measure) rather than by their formal names, because the operational properties are what carry the methodological argument; the formal names supply the implementation reference for the scoring procedure but are not required to follow the substantive discussion of the criteria.

3.5 The Closed Candidate List Design Pattern

The single most consequential design device in the framework is the closed-candidate-list pattern, applied to seven of the twelve tasks. The pattern transforms an open-ended elicitation task, in which the LLM produces free-text output that must be matched semantically against a ground truth, into a closed selection task, in which the LLM selects identifiers from a pre-defined list and scoring proceeds through direct set comparison on those identifiers. The transformation is the methodological device that closes the loop on the four commitments stated in Section 3.1: it preserves operationalisation through structured outputs, enables full automation of scoring, and aligns with the empirical validation logic of the second survey.

The pattern is applied to PM-01 and BA-01 (stakeholder identification from a roster of fifteen to twenty candidates), PM-02 (ranked picking of ten risks from a list of twenty), PM-04 (single choice among five WBS decomposition approaches), PM-05 (identification of five milestones from a list of fifteen), BA-03 (selection of explicit needs from a list of eighteen candidates), BA-04 (classification of fifteen pre-given requirements), and the Prioritize Requirements task (ranking of ten pre-given requirements). It is deliberately not applied to the Estimate

Costs task (numeric output), PM-03 (structured table with task-specific identifiers), the Sequence Activities task (relational output over a given activity set), or BA-02 (closed positioning of pre-given stakeholders). The pattern is therefore a primary device for selection-oriented tasks, while other output structures preserve their natural form.

Three properties of the closed-list pattern are essential to its methodological role. The first is the elimination of semantic-matching ambiguity. Without the pattern, scoring requires deciding whether an LLM output “CEO” matches a ground-truth “Chief Executive Officer”, whether “stakeholder management” matches “stakeholder engagement”, or whether “vendor” matches “supplier”. These decisions are intrinsically subjective, depend on context, and reintroduce expert judgement at the scoring stage. With the pattern, the LLM responds with the identifier S03 and scoring matches S03 against S03, with no interpretive degrees of freedom.

The second is the composition of the candidate list as a deliberate design object. Each candidate list contains a mixture of correct items, plausible distractors, and where relevant implicit or inferred items. The mixture is constructed so that the closed-list task remains cognitively meaningful: the LLM cannot simply select all candidates and accumulate true positives without paying the precision penalty, nor can it select none and avoid the recall penalty. The composition is calibrated to the ground-truth set size: BA-03, for example, presents eighteen candidates of which six are explicit needs, three are implicit-but-inferable, and nine are distractors, requiring the LLM to discriminate three categories in a single selection act.

The third is the alignment with second-survey validation logic. The second survey asks practitioners whether the proposed evaluation criterion is a sensible measure of the LLM’s accuracy on each task. The question is meaningful only if the criterion is automatable, since a criterion that requires expert rubric application is not a measure of LLM capacity but of the rubric itself. The closed-list pattern guarantees automatability, allowing the second survey to focus on the substantive methodological question of whether the criterion is a reasonable measure of professional accuracy rather than on the operational question of whether scoring can be performed at all.

The pattern is therefore not a convenience but a load-bearing methodological commitment. The literature reviewed in Chapter 2 contains no precedent for the systematic application of this pattern across a unified PM and BA taxonomy, although individual elements of it appear in benchmarks that compare LLMs on classification tasks (Wysocki & Ochodek, 2026; Erfani & Khanjar, 2025) and in studies that use closed candidate sets for stakeholder identification within a single task (Mircea et al., 2026). The generalisation of the pattern into a cross-task framework spanning both PM and BA processes is one of the methodological contributions of the present thesis.

3.6 Framework Deliverable: The Integrated Library

The framework consolidates the artefacts produced in the preceding sections into an integrated library that constitutes the central deliverable of the present thesis. The library comprises four components: a taxonomy of twelve tasks with their PMBOK and PMI Business Analysis coordinates (Section 3.2); a library of twelve operational questions, each paired with an input specification and an expected output structure (Section 3.3); a system of twenty-four evaluation criteria organised into three families of methodological trade-offs (Section 3.4); and a methodological commentary documenting the rationale of each design choice, presented throughout the chapter.

The library is methodologically complete in the sense that an LLM benchmark on PM and BA tasks can be

executed by following its specifications without further design work. Concretely, the execution of the benchmark proceeds in five steps. First, for each of the twelve tasks, an instance of the operational question is constructed by inserting a project description, a candidate list, and a ground truth into the question template; the template parameters are specified at the level of structural slots (for example, “twenty candidate stakeholders, eight relevant”) rather than at the level of specific content, so that multiple instances per task can be produced for variance estimation. Second, the LLM is prompted with the constructed instance and produces a structured output. Third, the output is parsed against the expected structure specified for the task; outputs that fail to parse are scored zero. Fourth, the parsed output is compared against the ground truth using the two evaluation criteria specified for the task. Fifth, the per-instance scores are aggregated over instances to produce per-task score estimates, with confidence intervals derived from the variance across instances.

This integrated library is the object of two complementary forms of empirical treatment in the chapters that follow. Its empirical validation is reported first, in Chapter 4: two surveys with practitioner respondents test whether the macro-processes are perceived as relevant for LLM benchmarking and whether the proposed evaluation criteria are perceived as methodologically valid, supplying the evidence on which the soundness of the framework rests before any model is scored against it. Its execution is reported next, in Chapter 5: the validated instrument is administered to a set of contemporary models, the scoring procedures are applied to their outputs, and the resulting cross-model differences are analysed.

The library is also extensible. The three families of methodological trade-offs are framework-level patterns rather than per-task particularities, and any further PM or BA task added in future work can be specified through the same template: a taxonomy coordinate from the PMBOK Guide or from the PMI Guide to Business Analysis, an operational question with a closed input and structured output, a candidate list constructed under the discipline of Section 3.5 where applicable, and two evaluation criteria each drawn from one of the three families. The extensibility of the framework is one of the central methodological commitments of the present work: the twelve tasks are illustrative cases, not a closed set, and the framework’s value resides in the design pattern it embodies as much as in the specific tasks it covers.

3.7 Operational Specification

The integrated library of the preceding section is complete as a design: it fixes the tasks, the operational questions, the evaluation criteria, and the scoring rules with sufficient precision that no further design choice is required to evaluate a model. A specification, however, is not yet an instrument. This section sets out, in summary, how the library is rendered executable; the protocol under which it is administered to models, and the results that administration produces, are the subject of Chapter 5.

Each task is realised as a fully specified item under the four design principles of Section 3.3: a prompt that instructs the model to return only the prescribed structure, a closed input artefact, a prescribed output structure, and an ex-ante ground truth constructed independently of any model run. Rather than evaluate each task on a single scenario, the framework evaluates it over a pool of instances, so that per-task scores are estimated with their dispersion rather than asserted from a single observation. The pool comprises nine hundred and ninety-eight instances produced by deterministic, seeded parametric templates, one per task, with the ground truth for

each instance computed by the same generative rules that produce it; the reference is therefore internal to the framework and depends on no external annotator, and the entire pool is auditable from its templates and seeds. The instances are allocated across the twelve tasks in proportion to the relevance that the corresponding macro-processes received in the first survey rather than in equal shares, carrying the survey’s empirical signal into the weight that each task receives at evaluation time, as Figure 3.1 depicts.

Each instance is presented under prompt configurations that hold the item fixed while varying the framing around it, ranging from a minimal instruction through a practitioner-framed variant and a reasoning-augmented variant to a stress variant in which the supporting material carries the noisy and partially contradictory detail of real project documents. Because the configurations differ only in the framing applied to identical items, any difference in score across them is attributable to the model’s robustness to framing rather than to the underlying task, which makes prompt sensitivity measurable as a first-class quantity. Scoring is performed by a deterministic pipeline that loads the pool, parses each returned output against the structure prescribed for its task (scoring unparseable outputs as null performance), compares the parsed output against the ground truth using the task’s official metric, and aggregates the per-instance scores into per-task estimates reported with their dispersion and separable by configuration and by model. Every official metric is an elementary arithmetic function of a structured output and a fixed ground truth, with no semantic matching and no human judgement at any point; the pipeline is reproducible and self-verifying, in that its scoring functions reproduce the worked examples documented for each criterion independently of any model run, and it is provided as a reproducible artefact in Appendix B.

The chapter that follows reports the empirical validation of the framework through two surveys with practitioners from the Project Management and Business Analysis communities.

Table 3.2: The twelve tasks of the framework, with taxonomy coordinates and derivation from the surveyed macro-processes.

ID	Type	Taxonomy coordinate	co-	Task name	Output structure	Macro-process
PM-01	PM	Stakeholders Initiating	×	Identify Stakeholders	List with attributes	Identify Stakeholders
PM-02	PM	Risk × Planning		Identify Risks	Ranked selection	Risk Management
PM-03	PM	Schedule Planning	×	Develop Schedule	Structured table	Scheduling
PM-04	PM	Scope × Planning		Plan Scope Management / Define Scope	Single choice	Planning
PM-05	PM	Integration Planning	×	Develop Project Management Plan	Ordered selection	Planning
PM-EXT-01	PM	Finance × Planning		Estimate Costs	Real positive number	Planning
PM-EXT-02	PM	Schedule Planning	×	Sequence Activities	Dependency graph	Scheduling
BA-01	BA	Stakeholder Eng. × Initiating		Identify Stakeholders (BA)	List with attributes	Identify Stakeholders
BA-02	BA	Stakeholder Eng. × Executing		Conduct Stakeholder Analysis	Two-by-two positioning	Analyze Stakeholders
BA-03	BA	Elicitation Executing	×	Conduct Elicitation	Selection from closed list	Elicit Needs
BA-04	BA	Analysis Planning	×	Define and Elaborate Requirements	Two-dimension classification	Baseline Requirements
BA-EXT-01	BA	Analysis Planning	×	Prioritize Requirements	Ranked list	Baseline Requirements

Table 3.3: The five output-structure classes of the framework.

Output class	Scoring approach	Tasks
Numeric	Relative error or tolerance band	Estimate Costs
Set selection	Selection accuracy (balanced over missed and wrong items)	PM-01, BA-01, BA-03, PM-05
Selection + classification	Selection accuracy plus attribute-classification accuracy	PM-01, BA-01, BA-04, BA-02
Ranked list	Rank-agreement measure (global, or weighted on top positions)	PM-02, Prioritize Requirements
Relational graph	Edge-set selection accuracy (with or without critical-path weighting)	Sequence Activities
Structured table	Feasibility check plus makespan ratio	PM-03

Table 3.4: Cognitive coverage of the twelve tasks, read through the levels of Bloom’s revised taxonomy.

Cognitive level	Task	Cognitive operation required
Apply	Estimate Costs	Execute an estimation procedure on a brief
Apply	Develop Schedule	Construct a feasible schedule under constraints
Apply	Sequence Activities	Derive the dependency relation among activities
Analyze	Identify Stakeholders (PM)	Distinguish relevant parties and characterise them
Analyze	Identify Stakeholders (BA)	Distinguish relevant parties and assign roles
Analyze	Identify Risks	Discriminate the salient risks from distractors
Analyze	Develop Project Management Plan	Identify the critical milestones and order them
Analyze	Conduct Stakeholder Analysis	Position parties on an influence-interest grid
Analyze	Conduct Elicitation	Separate explicit needs from inferred and unrelated ones
Analyze	Define and Elaborate Requirements	Classify requirements by type and priority
Evaluate	Plan Scope Management	Judge the most appropriate decomposition approach
Evaluate	Prioritize Requirements	Rank requirements by relative priority

Table 3.5: The twenty-four evaluation criteria of the framework, classified by family of trade-off.

Task	C#	Criterion	Family
PM-EXT-01	1	Relative absolute error	A
PM-EXT-01	2	Tolerance band ($\pm 10/20/30\%$)	A
PM-03	1	Feasibility (0.6) plus makespan efficiency (0.4)	B
PM-03	2	Feasibility gate plus makespan ratio	A
PM-01	1	Balanced selection accuracy on stakeholder set	B
PM-01	2	Selection accuracy (0.6) plus classification accuracy (0.4)	B
PM-02	1	Selection accuracy on top-10 risk set	B
PM-02	2	Ranked top-10 with top-positions-weighted measure (Järvelin & Kekäläinen, 2002)	B
BA-01	1	Selection accuracy (0.6) plus role-classification accuracy (0.4)	B
BA-01	2	Selection-only accuracy on stakeholder set	B
BA-04	1	Equal-weight classification accuracy (50/50)	C
BA-04	2	Priority-weighted classification (30/70)	C
PM-04	1	Discrete mapping with partial credit (1.0/0.5/0)	A
PM-04	2	Binary mapping (best-practice only)	A
PM-05	1	Selection accuracy (0.7) plus chronological order (0.3)	B
PM-05	2	Selection-only accuracy on milestone set	B
BA-02	1	Equal-weight positioning accuracy (50/50)	C
BA-02	2	Influence-weighted positioning (70/30)	C
BA-03	1	Symmetric selection accuracy on explicit-need set	C
BA-03	2	Selection accuracy with explicit/implicit penalty (0.5/1.0)	C
PM-EXT-02	1	Symmetric selection accuracy on dependency-edge set	C
PM-EXT-02	2	Critical-path-weighted edge-selection accuracy ($2\times$)	C
BA-EXT-01	1	Rank-agreement measure between full orderings (Spearman, 1904)	B
BA-EXT-01	2	Top-3 match (0.4) plus rank-agreement (0.6)	B

Chapter 4

Validation

The framework developed in Chapter 3 rests on a chain of design decisions: which professional processes are worth assessing, how each is turned into an operational task, and how the quality of a model's response to that task is measured. None of these decisions is self-evidently correct, and a framework whose only warrant is the judgement of its author offers a weak foundation for the benchmark that follows. This chapter establishes the empirical warrant. It reports two surveys of practitioners that filter the framework on two distinct dimensions and, taken together, ground its central choices in professional consensus rather than in design intuition.

4.1 Purpose and Strategy of the Validation

The validation pursued here is one of face and content validity: the degree to which domain experts judge that an instrument measures what it claims to measure and that its components are relevant and adequate for the construct of interest (Lawshe, 1975; Lynn, 1986). This is the appropriate standard for a methodological contribution at the design stage, where the object under examination is the framework itself rather than the performance of any model, and where the relevant evidence is the considered judgement of those who perform the work the framework seeks to represent. Content validity established through expert panels is a recognised and self-contained form of validation, distinct from the criterion or construct validity that a later application of the framework might additionally provide (Polit & Beck, 2006).

The validation is organised as two surveys that filter the framework on complementary dimensions. The first survey filters for professional relevance: it asks practitioners which Project Management and Business Analysis macro-processes are relevant and well suited to evaluation of model support, and its results define the priority map from which the twelve operational tasks are derived in Chapter 3. The second survey filters for methodological validity of measurement: it asks practitioners, for each task, whether a proposed scoring criterion is a sensible way to measure the accuracy of a model's response. The two surveys answer different questions and apply different filters, but they share a single methodological commitment, namely that the framework should be anchored in practitioner judgement at every level at which a discretionary choice was made.

Two cautions frame the interpretation of what follows and are developed further in Chapter 6. First, both sur-

veys recruit through a professional network and are therefore samples of convenience; the analysis is consequently a preliminary expert validation establishing face and content validity, not a statistically powered inferential study of a defined population. Second, at the sample sizes obtained, descriptive statistics carry the argument and any subgroup comparison is exploratory. These cautions do not weaken the validation for its stated purpose: expert-panel content validity is conventionally established with panels of comparable or smaller size (Lynn, 1986), and the convergence of the two surveys on the same design choices is itself the principal validation signal.

4.2 Method

4.2.1 Instruments

Both surveys were administered as anonymous online questionnaires in English, each opening with an informational statement of purpose, estimated duration, and anonymity. The first survey comprised twelve questions, three of them optional, and required approximately seven minutes; after a profiling section it elicited, on a one-to-five Likert scale, the relevance of seven PM and BA macro-processes, the formulation quality of the corresponding candidate tasks, the appropriateness of seven general and seven Business-Analysis-specific evaluation criteria, and the preferred scoring method, each rating question being accompanied by an optional free-text field. The second survey retained the same profiling section and then presented the twelve operational tasks one block at a time; each block stated the process, the concrete question that would be put to a model with an input example, and two candidate scoring criteria, each described in plain language with a worked numerical example. Respondents rated every criterion independently on the same one-to-five scale, answering in each case a single question: whether it makes sense to evaluate the accuracy of a model's response on that process through that criterion. With two criteria for each of the twelve tasks, the second instrument yielded twenty-four rated criteria. The scoring rules were presented to respondents using only elementary arithmetic and operational descriptions, so that the rating reflected the behaviour of each criterion rather than familiarity with any underlying statistical notation; the formal treatment of those techniques is confined to Chapter 3.

4.2.2 Respondents and Recruitment

Respondents to both surveys were practitioners recruited through a professional network on LinkedIn, Microsoft Teams, and electronic mail, targeting the Project Manager and Business Analyst communities while admitting adjacent roles. The first survey obtained forty complete responses and the second thirty-three; the two cohorts overlap only in part, so that the criteria validation is not merely a re-rating by the same individuals who established process relevance. The demographic composition of each sample is reported in the respective results section.

4.2.3 Analytical Approach and Decision Rules

Ratings on a one-to-five scale are treated as interval data for the purpose of computing means and standard deviations, a treatment that is well established for Likert items analysed in aggregate (Norman, 2010). For the first

survey, a macro-process is regarded as reaching core relevance when its mean relevance rating is at least 4.0; the threshold orders the macro-processes by professional priority rather than partitioning them into accepted and rejected sets, since the derivation of the operational tasks in Chapter 3 carries all surveyed macro-processes forward. For the second survey, a criterion is regarded as validated when its mean rating is at least 4.0. Because a fixed cut-off cannot, at this sample size, sharply separate a mean marginally below the line from one marginally above it, a criterion whose mean falls just short of 4.0 is also treated as validated when a one-sample test against the reference value 4.0 does not reject the hypothesis that its true mean is at least 4.0; that is, when its mean is not significantly below the threshold at the precision afforded by the sample. Validation establishes that a criterion is acceptable, not that it is the one to be used: where a task has two validated criteria, the higher-rated of the two is taken as the official metric and the other is retained as an acceptable alternative, and where a task has a single validated criterion that criterion becomes the official metric. Subgroup comparisons by professional role and experience band are reported as robustness checks and are interpreted as exploratory.

4.3 First Survey: Relevance of the Macro-Processes

4.3.1 Respondent Profile

The first survey obtained forty complete responses. Their composition across role, experience, and familiarity with generative-AI tools is reported in Table 4.1. The two target communities dominate the sample: Project Managers and Business Analysts together account for two-thirds of respondents, with the Project Management Office contributing a governance perspective and the remaining roles a smaller adjacent presence. Experience is distributed across all four bands with a mild concentration at the mid-level, and familiarity with generative-AI tools is high: nine respondents in ten report at least moderate hands-on use and none report no experience. The relevance judgements that follow can therefore be read as informed assessments by practitioners who both perform the work and use the technology, rather than as speculation.

4.3.2 Process Relevance

The relevance ratings of the seven macro-processes are shown in Figure 4.1. Identify Stakeholders leads the ranking with a mean of 4.30, followed by Risk Management at 4.18, with Planning and Baseline Requirements both at 4.00; four macro-processes thus reach core relevance. Elicit Needs and Analyze Stakeholders fall just below the threshold, at 3.93 and 3.80, while Scheduling is the only macro-process whose mean is clearly below the line, at 3.28.

The position of Scheduling deserves comment, because it carries the lowest relevance score yet enters the framework. A mean of 3.28 lies above the neutral midpoint, indicating qualified rather than absent professional interest, and the decision to retain it rests on a benchmarkability argument, its closed input, structured output, and objective ground truth, set out with the selection rationale in Section 3.2.1. The tension between its professional relevance and its benchmark value is taken up explicitly in Chapter 6.

Table 4.1: First survey — respondent profile (N = 40).

Dimension	Category	Count	Share
Role	Project Manager	14	35.0%
	Business Analyst	13	32.5%
	Project Management Office	7	17.5%
	Product Manager	3	7.5%
	Consultant	2	5.0%
	Other (outside target roles)	1	2.5%
Experience	0–2 years (junior)	7	17.5%
	3–5 years (mid)	15	37.5%
	6–10 years (senior)	11	27.5%
	>10 years (expert)	7	17.5%
GenAI familiarity	No experience	0	0.0%
	Limited	4	10.0%
	Moderate	12	30.0%
	High	24	60.0%

4.3.3 Relevance by Professional Role

Decomposing the relevance ratings by role reveals a pattern that the overall ranking conceals. Figure 4.2 compares the mean ratings of the three sizeable role groups across the seven macro-processes. The pattern is one of domain loyalty: Project Managers rate the PM-rooted processes highest, with Risk Management at 4.50 and Planning at 4.36, while Business Analysts rate the BA-rooted processes highest, with Elicit Needs at 4.92, Identify Stakeholders and Baseline Requirements both at 4.85, and Analyze Stakeholders at 4.38. The Project Management Office occupies an intermediate position, aligning with Project Managers on operational processes, where it rates Planning at 5.00 and Risk Management at 4.71, and diverging on the more analysis-oriented ones. The disagreement is substantial on several processes, exceeding two scale points between the highest- and lowest-rating community on Planning and on Elicit Needs.

This divergence is not a defect of the instrument but a substantive finding that supports the framework’s defining choice. The decision to integrate Project Management and Business Analysis processes in a single framework is vindicated precisely because each cluster of processes is endorsed by the community that performs it day to day: a framework confined to one discipline would have inherited that discipline’s blind spot toward the other’s priorities, understating the value of stakeholder and requirements work to Business Analysts or of risk and planning work to Project Managers. The overall ranking is therefore best understood not as a flat consensus but as the resolution of two coherent professional viewpoints, and the existence of the divergence motivates the role-segmented reading of the benchmark results interpreted in Chapter 6.

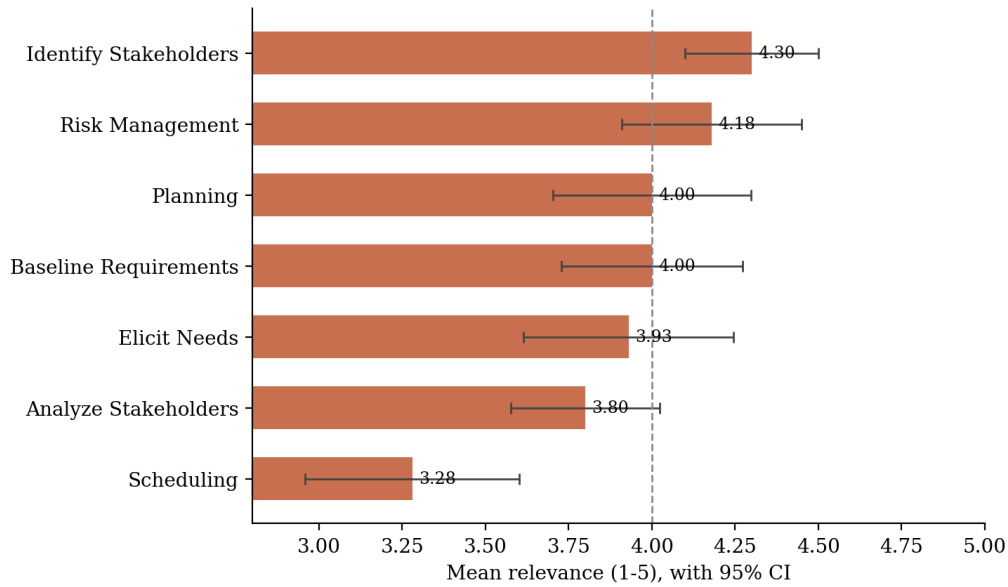


Figure 4.1: First survey — relevance of the seven macro-processes for model-supported evaluation, ordered by mean, with 95% confidence intervals. The dashed line marks the 4.0 core-relevance threshold.

4.3.4 Evaluation Criteria and Scoring Method

Beyond process relevance, the first survey elicited the appropriateness of two families of evaluation criteria and the preferred scoring method, the results of which directly shape the framework’s scoring architecture. Table 4.2 reports the means for the seven general criteria and the seven Business-Analysis-specific criteria. The general criteria are endorsed without exception, all scoring above 4.3, with the technical-correctness dimensions leading: Accuracy and Logical coherence both at 4.80, Adherence to provided context at 4.78, and Absence of fabricated information at 4.75. The two lowest, Decision usefulness at 4.35 and Clarity of expression at 4.33, remain well above neutral but are seen as somewhat secondary to factual correctness, a hierarchy that informs the weighting of the general-quality family in the scorecard. The Business-Analysis-specific criteria show a wider spread. Requirements clarity dominates at 4.93, followed by a cluster comprising Coverage of stakeholders, Consistency between need and requirement, and Identification of interests and conflicts, all near 4.55. The two lowest, Identification of explicit needs at 3.90 and Uncovering of plausible implicit needs at 3.75, are themselves informative: the modest rating of implicit-need uncovering provides independent support for the framework’s decision to restrict the elicitation task to explicit needs, since practitioners themselves regard the uncovering of implicit needs as the more uncertain dimension.

The question on scoring method produced the survey’s most directive result. A combination of multiple methods was preferred by 72.5% of respondents (29 of 40), far ahead of a single analytical rubric at 15.0% (6), a qualitative expert assessment at 10.0% (4), and a single numerical scale at 2.5% (1). This expressed preference for multi-method evaluation is the empirical basis for one of the framework’s structural choices, namely that each task is assessed through two independent criteria rather than one, and that the scoring combines a structured rubric with an expert-judgement component. The free-text responses reinforced and refined this direction: recurring requests for evaluation under both sparse and rich prompts, for assessment of multi-step reasoning quality, and for

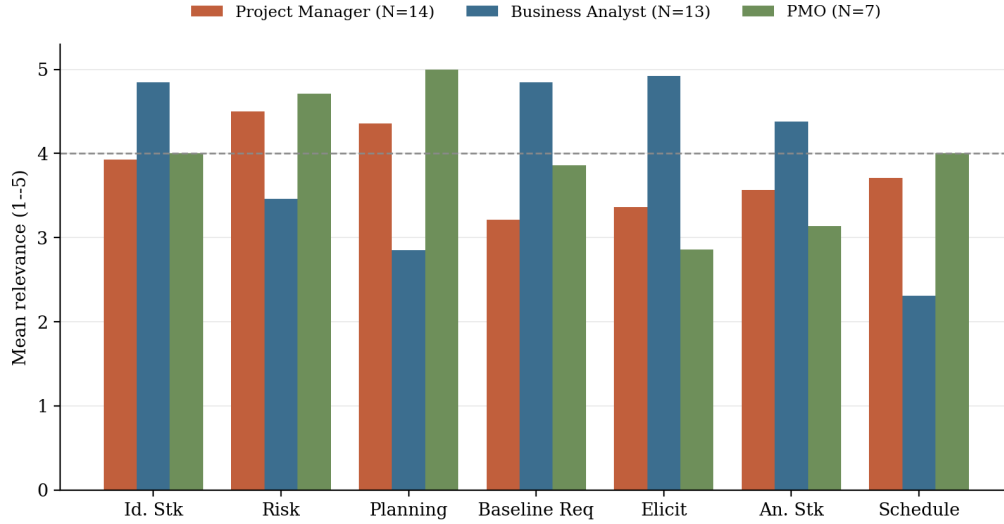


Figure 4.2: First survey — mean relevance rating of each macro-process by professional role. The dashed line marks the 4.0 threshold.

Table 4.2: First survey — appropriateness of the evaluation criteria (mean rating, N = 40).

General criteria		BA-specific criteria	
Criterion	Mean	Criterion	Mean
Accuracy	4.80	Requirements clarity	4.93
Logical coherence	4.80	Stakeholder coverage	4.55
Adherence to context	4.78	Need–requirement consistency	4.55
No fabricated information	4.75	Interests and conflicts	4.53
Completeness	4.58	Need–requirement traceability	4.13
Decision usefulness	4.35	Explicit needs	3.90
Clarity of expression	4.33	Implicit needs	3.75

stress testing with incomplete or contradictory inputs map onto the prompt configurations of the benchmark, while requests for explicit treatment of assumption transparency and refusal to fabricate inform the robustness family of criteria. The qualitative material thus contributes not isolated opinions but a coherent set of design directions, each traceable to a countable number of respondents.

4.4 Second Survey: Validity of the Evaluation Criteria

Where the first survey filtered the processes, the second filters the measurements. It subjects each of the twenty-four candidate criteria to the same single question of appropriateness and, through the decision rule of Section 4.2.3, determines for each task the criterion that will serve as its official scoring metric.

4.4.1 Respondent Profile

The second survey obtained thirty-three complete responses, with no missing values across the twenty-four rated criteria, so that every criterion is analysed on the full sample. Table 4.3 reports the composition. Project Managers and Business Analysts again form the core, jointly just under three-quarters of the sample, and the experience distribution places eighteen respondents in the senior or expert bands. Familiarity with generative-AI tools is again high, with thirty of thirty-three reporting at least occasional professional use. Thirteen respondents had also taken part in the first survey and twenty had not, so that the criteria validation draws substantially on practitioners who were not involved in establishing process relevance; the second-survey instrument is reproduced in Appendix C.

Table 4.3: Second survey — respondent profile (N = 33).

Dimension	Category	Count	Share
Role	Project Manager	13	39.4%
	Business Analyst	11	33.3%
	Project Management Office	4	12.1%
	Consultant	4	12.1%
	Product Manager	1	3.0%
Experience	0–2 years (junior)	5	15.2%
	3–5 years (mid)	10	30.3%
	6–10 years (senior)	10	30.3%
	>10 years (expert)	8	24.2%
GenAI familiarity	Never (at work)	3	9.1%
	Occasional	10	30.3%
	Regular	12	36.4%
	Advanced	8	24.2%

4.4.2 Validation Results

The appropriateness ratings of the twenty-four criteria are shown in Figure 4.3 and reported in full in Table 4.4. The headline result is that the criteria set is broadly endorsed. Thirteen of the twenty-four criteria reach a mean of at least 4.0; the remaining eleven fall within or just below the tolerance band, and in every case the 95% confidence interval of the mean includes the threshold. The one-sample test confirms this reading: no criterion is significantly below 4.0. Under the decision rule of Section 4.2.3, every criterion is therefore validated, and the analysis turns to selecting, for each task, the criterion that will serve as its official metric.

4.4.3 Within-Process Comparison and Metric Selection

Selecting the official metric for each task requires comparing its two candidate criteria directly, which Figure 4.4 does. For most tasks the gap between the two candidates is small, confirming that both framings are reasonable,

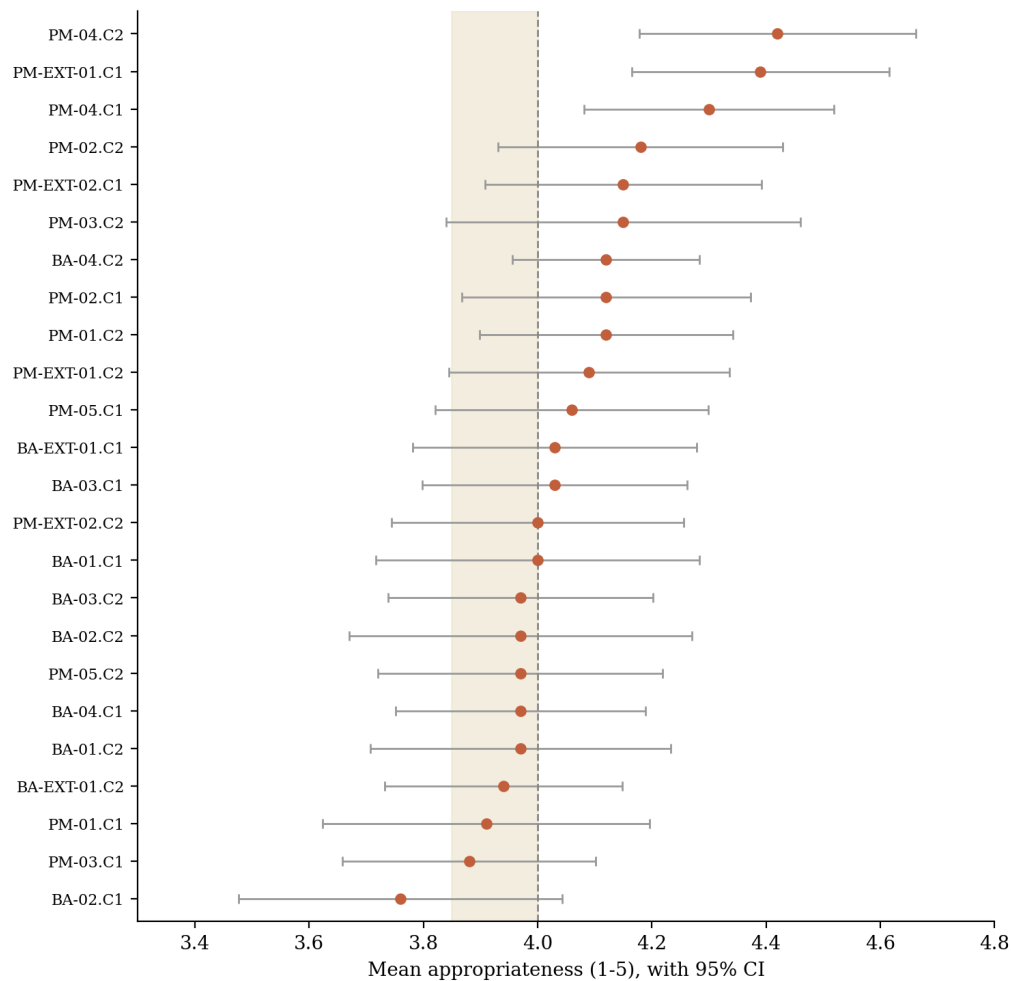


Figure 4.3: Second survey — mean appropriateness rating of the twenty-four criteria, ordered by mean, with 95% confidence intervals. The dashed line marks the 4.0 threshold and the shaded band the 3.85–4.0 tolerance zone.

but the direction of the preference is informative and consistent with professional reasoning. The stricter or more judgement-intensive variant is preferred where the consequences of a coarse measure are greatest: the strict feasibility gate for Develop Schedule, the priority-weighted classification for Define Requirements, the best-practice-only rule for Plan Scope, and the influence-weighted positioning for Conduct Stakeholder Analysis. The simpler selection-only or proportional variant leads where the additional structure adds little: the proportional estimate for Estimate Costs, the equal-weight dependency measure for Sequence Activities, and the global rank-agreement measure for Prioritize Requirements.

4.4.4 Consensus and Dispersion

A mean rating can arise from broad agreement or from a polarised split, and the two carry different implications for the confidence with which a criterion can be adopted. The standard deviations reported in Table 4.4 distinguish the two cases. Endorsement is generally tight: most criteria carry a dispersion between 0.6 and 0.8 on the five-point scale, and the single closest agreement falls on a validated criterion (Define Requirements C2, mean 4.12,

Table 4.4: Second survey — validation results for the twenty-four criteria (N = 33). Verdict per the decision rule of Section 4.2.3.

Code	Process	Crit.	Mean	SD	Verdict
PM-EXT-01	Estimate Costs	C1	4.39	0.66	Validated (≥ 4.0)
		C2	4.09	0.72	Validated (≥ 4.0)
PM-03	Develop Schedule	C1	3.88	0.65	Validated (margin)
		C2	4.15	0.91	Validated (≥ 4.0)
PM-01	Identify Stakeholders (PM)	C1	3.91	0.84	Validated (margin)
		C2	4.12	0.65	Validated (≥ 4.0)
PM-02	Identify Risks	C1	4.12	0.74	Validated (≥ 4.0)
		C2	4.18	0.73	Validated (≥ 4.0)
BA-01	Identify Stakeholders (BA)	C1	4.00	0.83	Validated (≥ 4.0)
		C2	3.97	0.77	Validated (margin)
BA-04	Define Requirements	C1	3.97	0.64	Validated (margin)
		C2	4.12	0.48	Validated (≥ 4.0)
PM-04	Plan Scope	C1	4.30	0.64	Validated (≥ 4.0)
		C2	4.42	0.71	Validated (≥ 4.0)
PM-05	Develop PM Plan	C1	4.06	0.70	Validated (≥ 4.0)
		C2	3.97	0.73	Validated (margin)
BA-02	Stakeholder Analysis	C1	3.76	0.83	Validated (margin) [†]
		C2	3.97	0.88	Validated (margin)
BA-03	Conduct Elicitation	C1	4.03	0.68	Validated (≥ 4.0)
		C2	3.97	0.68	Validated (margin)
PM-EXT-02	Sequence Activities	C1	4.15	0.71	Validated (≥ 4.0)
		C2	4.00	0.75	Validated (≥ 4.0)
BA-EXT-01	Prioritize Requirements	C1	4.03	0.73	Validated (≥ 4.0)
		C2	3.94	0.61	Validated (margin)

[†] Boundary case: lowest-rated criterion in the instrument; one-sided $p \approx 0.05$ against the 4.0 reference. Not selected as the official metric (see Section 4.4.5).

SD = 0.48). The widest dispersions, around 0.9, attach to Develop Schedule C2 and Conduct Stakeholder Analysis C2, where a larger share of neutral ratings reflects uncertainty about framing rather than disagreement with the measurement. No criterion combines a low mean with a tight cluster of low ratings, which is the signature a genuinely rejected criterion would leave; the marginal cases are therefore marginal in confidence rather than contested, a reading the free-text evidence analysed below corroborates.

4.4.5 The Single Decision Point

Only one task lacks a criterion above 4.0, and it is the sole point at which the decision rule must adjudicate rather than confirm. For Conduct Stakeholder Analysis, the influence-weighted variant attains a mean of 3.97 with a confidence interval that comfortably includes the threshold, and a one-sample test offers no basis for declaring it sub-threshold; it is, moreover, the higher-rated of the two candidates. It is therefore validated and adopted as the official metric for the task. The equal-weight variant, at 3.76, is the lowest-rated criterion in the entire instrument

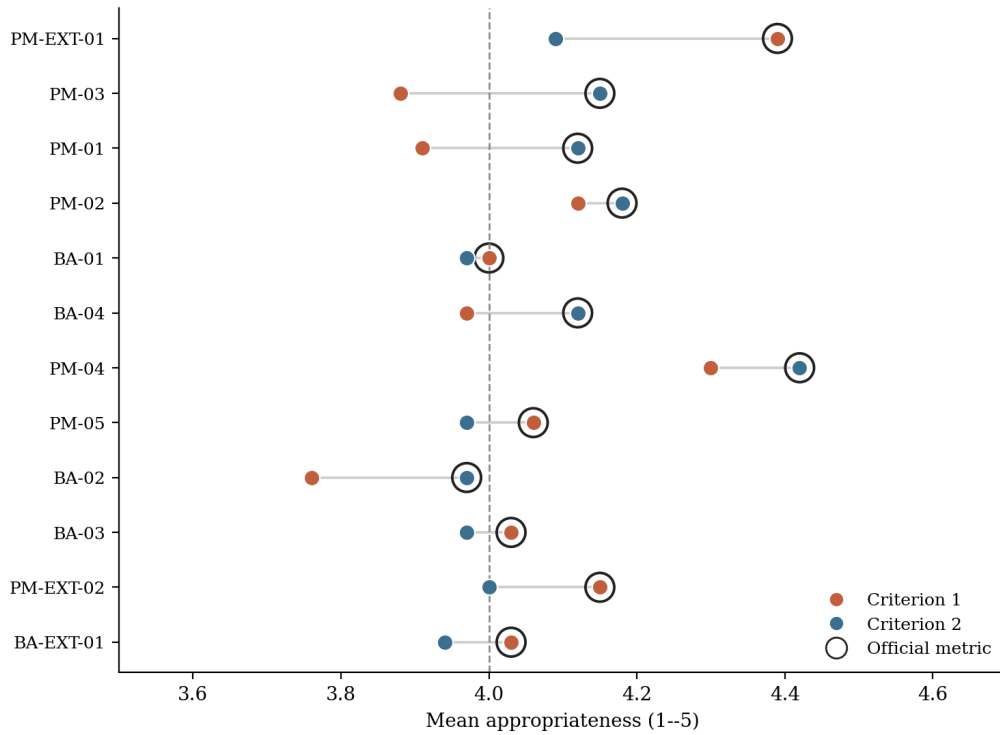


Figure 4.4: Second survey — the two candidate criteria for each task. The circled marker is the higher-rated criterion, adopted as the task’s official metric.

and the only one whose one-sided probability against the reference value approaches the conventional five-percent mark; while the tolerance rule does not declare it sub-threshold, it is not selected as the official metric, since the task already possesses a stronger validated criterion. This resolution preserves a validated scoring rule for every task while assigning the one genuinely marginal criterion its appropriate weight. A free-text comment on this task, observing that influence levels are highly context-dependent and that a coarse partition is unsatisfactory, bears on the construct rather than on the choice between the two weightings and is carried into the limitations discussion of Chapter 6.

4.4.6 Robustness

Two checks confirm that the validation does not depend on the composition of the sample. Across experience bands, the per-respondent mean rating over all twenty-four criteria is stable and close to the overall average of 4.06, with no band systematically more or less generous; more experienced respondents show somewhat tighter dispersion, a sign of more consistent judgement rather than of a different level. The role-level reading points the same way. Averaged across the twenty-four criteria, the overall endorsement is almost identical for the three sizeable role groups, at 4.04 for Project Managers, 4.03 for Business Analysts, and 4.04 for the Project Management Office, with Consultants marginally higher. The absence of a role effect at the aggregate level is itself a validation signal, indicating that the criteria are accepted across communities rather than carried by one; role-specific preferences appear only in the within-task direction of Figure 4.4, not in the overall level of endorsement.

4.4.7 Convergence of the Qualitative Evidence

The optional comments, though sparse, converge with the quantitative pattern in a way that strengthens the metric choices. Several remarks argue against all-or-nothing scoring, observing that scheduling quality is gradual rather than binary and that scope definition is rarely binary so that partial credit is more realistic; others insist that classification and prioritisation matter alongside mere selection, and that requirement priority generally outweighs requirement type in real decisions. In each instance the quantitative data already favoured the variant the comment points to, so that the qualitative and quantitative evidence corroborate rather than merely coexist. On processes outside the present scope, respondents most often named Quality Management, Procurement, project closing and lessons learned, performance and status reporting, solution evaluation, and conflict management; these are useful for positioning the boundaries of the framework in Chapter 6 rather than for the present validation. Two methodological observations recur and are worth carrying forward: sensitivity to prompt phrasing, and the consistency of a model across repeated runs of the same prompt, the latter proposed explicitly as a meta-criterion.

4.5 Synthesis: The Validated Framework

The two surveys, taken together, supply the empirical warrant the framework requires. The first establishes that the macro-processes are professionally relevant, that their relevance is structured by a coherent division between the Project Management and Business Analysis communities that vindicates the integrated design, and that practitioners prefer multi-method evaluation, a structured rubric combined with expert judgement, over any single scoring approach. The second establishes that the proposed measurements are judged appropriate across the board and identifies, for each task, the criterion to be used as its official metric. Table 4.5 consolidates this outcome into the scoring layer that the framework carries forward.

Two of these selections coincide with metric choices already fixed during the framework design, namely the top-positions-weighted variant for Identify Risks and the global rank-agreement measure for Prioritize Requirements; for these the survey provides independent practitioner support rather than a revision. The remainder represent fresh empirical evidence that fixes the operational scoring rule for each task. With the processes validated for relevance, the measurements validated for appropriateness, and the single marginal case resolved under an explicit rule, the framework is empirically complete. Its implications for the framework's standing and the limits of what has been demonstrated are taken up in Chapter 6, while the comparative execution that the validated framework makes possible, the administration of the instrument to contemporary models and the analysis of the resulting scores, is reported in Chapter 5.

Table 4.5: Validated scoring layer — the official metric adopted for each task.

Code	Process	Metric	Official scoring rule
PM-EXT-01	Estimate Costs	C1	Estimate accuracy, proportional to deviation
PM-03	Develop Schedule	C2	Strict feasibility gate plus duration ratio
PM-01	Identify Stakeholders (PM)	C2	Selection accuracy plus attribute classification (60/40)
PM-02	Identify Risks	C2	Selection with top-positions-weighted ranking
BA-01	Identify Stakeholders (BA)	C1	Selection accuracy plus role classification (60/40)
BA-04	Define Requirements	C2	Classification accuracy, priority weighted (30/70)
PM-04	Plan Scope	C2	Strict scoring, best-practice answer only
PM-05	Develop PM Plan	C1	Selection accuracy plus chronological ordering (70/30)
BA-02	Stakeholder Analysis	C2	Positioning accuracy, influence weighted (70/30)
BA-03	Conduct Elicitation	C1	Selection accuracy, symmetric penalty
PM-EXT-02	Sequence Activities	C1	Dependency accuracy, equal weight on all edges
BA-EXT-01	Prioritize Requirements	C1	Rank-agreement measure, global ordering similarity

Chapter 5

Benchmark Execution and Results

The preceding chapters designed a framework for evaluating Large Language Models on Project Management and Business Analysis processes and established its empirical warrant through two practitioner surveys. The framework is, at the close of Chapter 4, a validated instrument: its processes are endorsed for relevance, its measurements for appropriateness, and its scoring layer fixed task by task in Table 4.5. This chapter applies the instrument. It administers the twelve benchmark tasks to two contemporary general-purpose models over the full instance pool, scores every returned answer through the deterministic pipeline using the official metric validated for each task, and reports the resulting comparison. In doing so it answers the fourth research question of the thesis: whether the validated framework, once executed, reveals interpretable differences in task-level and domain-level performance and produces a meaningful and reproducible spread of scores across models of differing capability.

The chapter is organised as follows. Section 5.1 states which models are evaluated and why. Section 5.2 describes the administration protocol under which the instance pool was presented to each model. Section 5.3 describes the scoring procedure and the discipline under which model outputs were parsed and compared against the ground truth. Section 5.4 reports the aggregate results at the level of discipline and overall, Section 5.5 the full per-task results, and Section 5.6 interprets the task-level pattern. Section 5.7 reports the checks that protect the integrity of the comparison, and Section 5.8 returns to the fourth research question.

5.1 Models Under Evaluation

The execution evaluates two widely used contemporary conversational models, Claude Sonnet 4.6 and ChatGPT GPT-5.5 Thinking (Standard), each accessed through its publicly available consumer interface during the administration window. The choice of two models from distinct providers is deliberate and proportionate to the purpose of the chapter, which is to demonstrate the instrument in use rather than to compile an exhaustive market ranking. Two models of differing capability are sufficient to test the central property an evaluation instrument must possess, namely that it separates stronger from weaker performance in an interpretable manner; a larger panel would multiply the administration effort without changing the methodological claim the chapter advances. The standard consumer interfaces were used for both models so that the comparison holds the access level constant and reflects

the configuration in which the majority of practitioners encounter these systems.

One further model was considered and excluded for a documented reason. Gemini was not administered because its consumer interface did not reliably accept the batched input artefacts under the administration protocol described in Section 5.2, so that a comparison holding the protocol constant across all models could not be guaranteed; including it under a different administration regime would have confounded model capability with administration method and undermined the very comparability the protocol is designed to secure. The exclusion does not weaken the demonstration: the instrument’s discriminating behaviour is established by the two models that were administered under the identical protocol, and extending the comparison to a broader set of models is a matter of repeating that protocol rather than of redesigning the instrument, as Chapter 7 notes.

5.2 Administration Protocol

Both models were administered the same instance pool under a single fixed protocol between 10 and 12 June 2026. The pool comprises the nine hundred and ninety-eight instances generated by the parametric templates described in Section 3.7 and reported in full in Appendix D, distributed across the twelve tasks in proportion to first-survey relevance. Each task block was presented to the model as a batch through a structured input attachment, accompanied by the canonical task prompt, the practitioner-framed configuration that directs the model to read the supporting material as an experienced PM and BA practitioner before answering and to return only the prescribed output structure with identifiers drawn from the closed candidate list where the task uses one. The prompt and the administration regime were identical across the two models, so that any difference in their scores is attributable to model behaviour rather than to a difference in how the task was posed.

One asymmetry arose during administration and is reported here because it is part of the protocol rather than a deviation from it. In its first pass, ChatGPT truncated its output on seventy-five instances concentrated in two task blocks, returning fewer answers than the block contained because the response reached an output-length limit before the block was complete. These seventy-five instances were re-administered on 12 June in a single second pass using the identical prompt and protocol, and all seventy-five were returned; the model therefore reached full coverage of nine hundred and ninety-eight instances. Claude completed every block in a single pass and required no re-administration. The second pass is declared as part of the administration record rather than concealed, both because reproducibility requires it and because the difference in output-completion behaviour under batch administration is itself a mild and honestly reportable signal: the weaker model needed a second presentation to deliver a complete answer set on the longer blocks, while the stronger model did not. The instances re-administered on the second pass are flagged in the per-item record that accompanies the thesis.

5.3 Scoring Procedure

Every returned answer was scored by the deterministic pipeline documented in Appendix B, using for each task the official metric fixed in Table 4.5. The pipeline loads the instance pool, parses each model output against the structure prescribed for its task, compares the parsed output against the ex-ante ground truth using the task’s

official metric, and aggregates the per-instance scores into per-task estimates. No step of the procedure involves semantic matching, expert rubric application, or human judgement: each official metric is an elementary arithmetic function of a structured output and a fixed reference, so that the same outputs scored under the same metrics yield the same scores on any run.

Three properties of the scoring discipline are essential to the integrity of the comparison and are stated explicitly because they bear directly on how the results should be read. The first is that parsing is model-agnostic: the rules that map a model's raw output onto the structured form the scorer expects were applied identically to both models, with no model-specific cleaning. Where a model returned an answer in a recognised format variant, a deterministic normalisation rule recovered it for both models alike; the two cases that arose were bare numeric rows in the Develop Schedule block, recovered for sixty ChatGPT outputs, and a candidate-description fallback in the Conduct Elicitation block, which mapped thirty-six ChatGPT outputs onto their candidate identifiers. The second property is that deviations which could not be normalised deterministically were scored as model errors rather than repaired by hand. Two such patterns occurred, both in ChatGPT outputs: the return of an invented identifier in place of the prescribed answer key on twelve Plan Scope instances, and the return of free-text names under an attribute schema different from the one the task prescribed on nineteen Identify Stakeholders (BA) instances. These were not corrected, because fidelity to the prescribed output structure is part of what the task measures: a model that does not answer in the form the task specifies has, to that extent, not performed the task, and the framework's commitment to structured outputs (Section 3.3) would be hollow if it were quietly waived at scoring time. The third property is that the pipeline is self-verifying: its scoring functions reproduce the worked examples documented for each criterion, so that the correctness of the implementation can be checked independently of any model run.

5.4 Aggregate Results

At the aggregate level the two models are clearly separated. Averaged across all twelve tasks, Claude Sonnet 4.6 attains a mean official-metric score of 0.807 and ChatGPT a mean of 0.545, a difference of twenty-six points on the unit scale. The separation is present in both disciplines but is wider on the Project Management side: across the seven PM tasks Claude averages 0.849 against ChatGPT's 0.544, a gap of thirty points, while across the five BA tasks Claude averages 0.748 against ChatGPT's 0.546, a gap of twenty points. Figure 5.1 reports the aggregate comparison by discipline and overall.

Two features of the aggregate picture are worth stating before the per-task detail. The first is that the stronger model's advantage is not uniform across the two disciplines: Claude's own performance is markedly higher on the more structured Project Management tasks (0.849) than on the more judgement-laden Business Analysis tasks (0.748), whereas ChatGPT performs at almost the same level on the two sides (0.544 and 0.546). The disciplines therefore differ in how strongly they reward additional capability, with the structured PM tasks separating the two models more sharply than the BA tasks. The second is that the overall gap of twenty-six points is large enough to be unambiguous yet arises from a profile of task-level differences that range from near-coincidence to near-total separation, which is the property the per-task analysis examines.

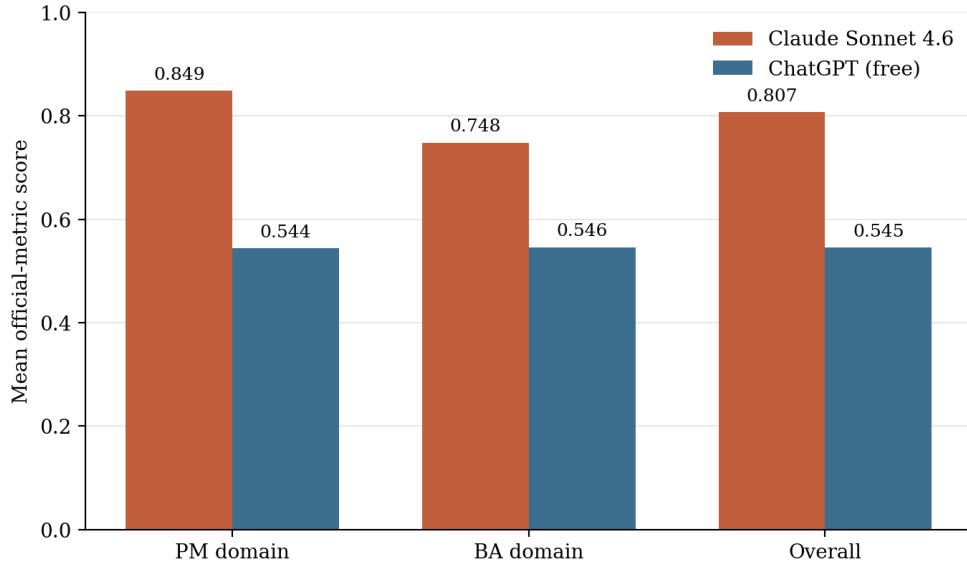


Figure 5.1: Mean official-metric score by discipline and overall, for the two models across the full instance pool of 998 instances.

5.5 Per-Task Results

Table 5.1 reports, for each of the twelve tasks, the number of instances, the official metric, the mean score of each model, and their difference. Figure 5.2 presents the same per-task scores on the twelve-axis profile whose allocation was established in Figure 3.1, so that the observed performance is read against the structure of the framework.

Claude attains the higher score on eleven of the twelve tasks. The single exception is Develop Schedule (PM-03), where ChatGPT reaches a perfect 1.000 against Claude’s 0.967, a margin of three points. On the remaining eleven tasks the difference favours Claude and ranges from twelve points (Prioritize Requirements) to forty-seven points (Develop PM Plan). The profile is therefore not a uniform offset but a structured pattern in which the two models coincide on one task, differ modestly on several, and separate almost completely on others.

5.6 Task-Level Analysis

The task-level pattern is interpretable along a single axis: the degree to which the task admits a single, structurally constrained correct answer as opposed to a graded judgement over a candidate space.

At one end stand the tasks on which the two models converge. Develop Schedule, the only task ChatGPT wins, is the most tightly constrained task in the framework: the model must assign start and end days under explicit duration and dependency constraints, and the official metric gates a feasible schedule before scoring its efficiency. A task with this much structure admits little room for a capability gap to express itself, and both models solve it at or near ceiling. Conduct Elicitation, on which the gap is the second-smallest after Prioritize Requirements, behaves similarly: the closed selection of explicit needs from a candidate list is the cleanest selection task in the framework, and Claude scores it perfectly while ChatGPT remains high. Where the task reduces to recovering a

Table 5.1: Per-task benchmark results over the full instance pool. Score is the mean of the task’s official metric (Table 4.5) across the task’s instances.

Code	Task	N	Metric	Claude	ChatGPT	Δ
PM-01	Identify Stakeholders (PM)	78	C2	0.842	0.669	+0.174
PM-02	Identify Risks	152	C2	0.646	0.293	+0.353
PM-03	Develop Schedule	60	C2	0.967	1.000	−0.033
PM-04	Plan Scope (WBS strategy)	48	C2	0.917	0.500	+0.417
PM-05	Develop PM Plan (milestones)	48	C1	0.918	0.444	+0.475
PM-EXT-01	Estimate Costs	49	C1	0.795	0.468	+0.327
PM-EXT-02	Sequence Activities	59	C1	0.857	0.436	+0.421
BA-01	Identify Stakeholders (BA)	78	C1	0.842	0.408	+0.434
BA-02	Conduct Stakeholder Analysis	138	C2	0.568	0.420	+0.148
BA-03	Conduct Elicitation	143	C1	1.000	0.863	+0.137
BA-04	Define Requirements	72	C2	0.513	0.337	+0.176
BA-EXT-01	Prioritize Requirements	73	C1	0.818	0.701	+0.116
PM domain mean				0.849	0.544	+0.305
BA domain mean				0.748	0.546	+0.202
Overall				0.807	0.545	+0.262

well-defined structure from an unambiguous input, additional model capability has little to add, and the instrument correctly registers the convergence rather than manufacturing a difference.

At the other end stand the tasks on which the models separate most sharply, and these are tasks that combine an ordered or attribute-laden output with the discipline of answering in a prescribed form. Develop PM Plan, where the gap reaches forty-seven points, requires selecting the critical milestones and ordering them chronologically; Sequence Activities, at forty-two points, requires recovering a dependency relation; Plan Scope, at forty-two points, requires identifying the single best-practice decomposition under a strict rule; and Identify Stakeholders (BA), at forty-three points, requires selecting stakeholders and assigning them a role schema. On these tasks ChatGPT’s lower score reflects two distinct shortfalls that the framework does not separate but that are worth distinguishing in interpretation. The first is genuine capability: ranking, dependency recovery, and best-practice judgement are exactly the operations the framework places at the Analyze and Evaluate bands of Table 3.4, and a weaker model loses ground on them. The second is fidelity to the prescribed output structure, which on Plan Scope and Identify Stakeholders (BA) ChatGPT violated on a non-trivial number of instances by returning an invented answer key or a non-conforming attribute schema. As Section 5.3 records, these were scored as errors rather than repaired, because answering in the specified form is part of the task; the consequence is that part of ChatGPT’s deficit on these tasks is a deficit of instruction-following rather than of domain reasoning. The framework is, by design, indifferent to which of the two a low score reflects, since a benchmark of operational task performance must hold a model to the form of the answer as well as to its content, but the distinction is available to a reader interpreting the comparison.

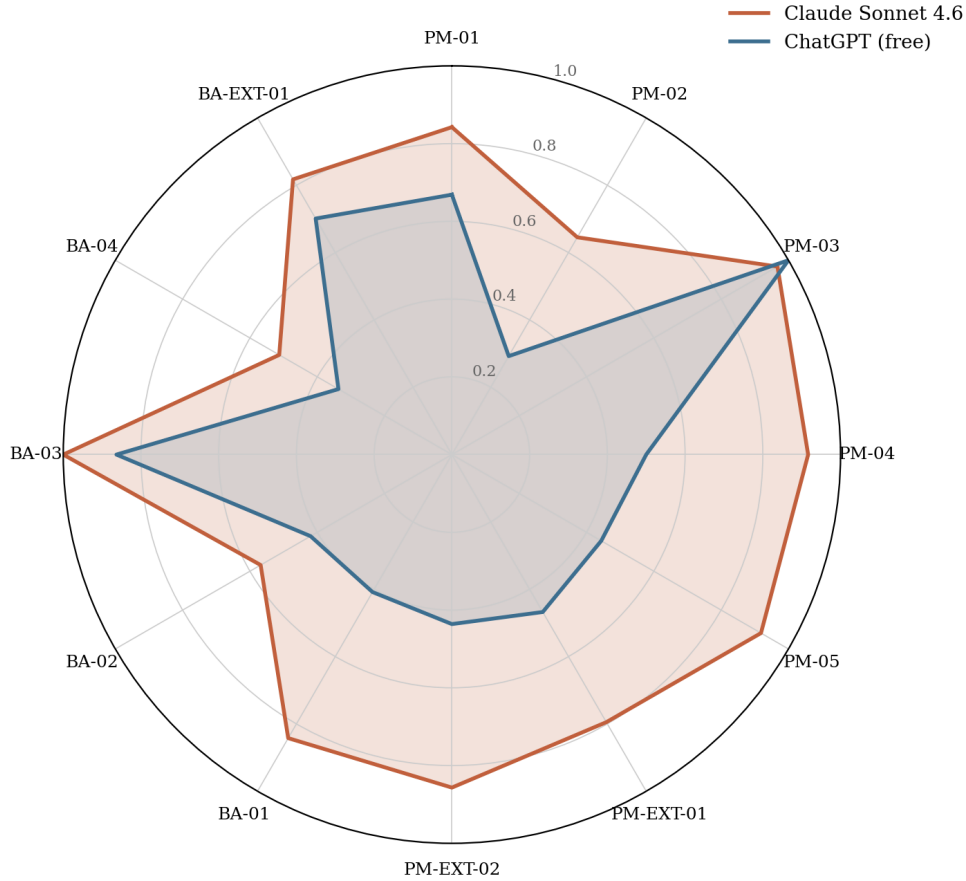


Figure 5.2: Per-task official-metric scores for the two models across the twelve benchmark tasks. The axes are those of Figure 3.1; the radial coordinate is the mean official-metric score on the unit interval.

Between these extremes sit the most demanding tasks for both models. Define Requirements is the hardest task in the framework: even the stronger model reaches only 0.513, and the weaker 0.337. The task requires classifying pre-given requirements on two dimensions simultaneously, type and priority, under a priority-weighted metric, and the low ceiling for both models indicates that the task is genuinely difficult rather than that the metric is miscalibrated. Conduct Stakeholder Analysis, the other low-ceiling task, behaves the same way, with both models in the middle of the scale. That the hardest tasks depress both models rather than only the weaker one is itself evidence that the instrument measures task difficulty and not merely a fixed model ranking: a metric that simply separated the two models by a constant would not produce the joint compression observed on these two tasks.

Identify Risks deserves separate mention because it carries the largest share of the instance pool, at one hundred and fifty-two instances, and because it is the task on which the gap is widest among the heavily populated blocks. Claude attains 0.646 and ChatGPT 0.293 on the ranked selection of the ten most salient risks from a candidate list of twenty, scored under a top-positions-weighted measure. The weaker model's score near 0.29 indicates that it both misses salient risks and orders the ones it selects poorly, two failures that the top-positions-weighted metric compounds. Because this block is the most heavily weighted in the pool by the relevance-proportional allocation of Section 3.2.6, the gap on it contributes substantially to the aggregate PM advantage, which is appropriate: the framework concentrates resolution where practitioners judged relevance to be highest,

and the model difference is correspondingly most consequential there.

5.7 Integrity of the Comparison

Three checks protect the comparison from the most common artefacts that distort head-to-head LLM evaluations. The first concerns coverage: both models were scored on the complete pool of nine hundred and ninety-eight instances, with no instance dropped for either model and no task block partially covered, so that the aggregates are computed over identical denominators and the per-task means are not inflated by selective omission. The second concerns the second pass: because seventy-five ChatGPT instances were re-administered, it is necessary to confirm that the re-administration did not advantage the model, and the per-item record shows that the re-administered instances are distributed across their blocks at scores consistent with the first-pass instances of the same blocks rather than systematically higher, so that the second pass restored coverage without improving the model's standing.

The third check concerns answer convergence. On seventy-seven first-pass instances ChatGPT returned an answer byte-identical to Claude's, which on a superficial reading might suggest contamination or copying. Inspection shows the convergence to be legitimate and concentrated on objective-answer tasks: all thirty-six identical answers in the Conduct Elicitation block are equal to the ground truth itself, which is precisely what two competent models should produce on a task that admits a single correct selection. Identity of answers on a task with a unique correct answer is convergence on the truth, not evidence of a shared source; it is the expected behaviour of two capable models on the cleanest tasks, and its concentration on exactly those tasks confirms rather than threatens the integrity of the comparison.

5.8 Synthesis

The execution answers the fourth research question. Applied to two contemporary models of differing capability, the validated framework produces a clear and interpretable spread of scores: an overall separation of twenty-six points, a per-task profile that ranges from near-coincidence on the most constrained task to a forty-seven-point gap on the most demanding ordered-selection task, and a domain-level pattern in which the more structured Project Management tasks separate the models more sharply than the more judgement-laden Business Analysis tasks. The spread is not an artefact of a permissive metric: the hardest tasks compress both models toward the middle of the scale, the easiest separate them only slightly, and the single task ChatGPT wins is the one whose structure leaves least room for capability to express itself. The instrument therefore behaves as an evaluation instrument should, registering convergence where the task admits a single answer, separation where the task rewards capability, and difficulty where the task is hard for any model.

The execution also makes good on the scoring discipline the framework committed to in its design. Every score in this chapter is the output of a deterministic arithmetic comparison between a structured model output and an ex-ante ground truth, applied identically to both models, with format normalisation applied alike and non-normalisable deviations counted as errors rather than repaired. The comparison is reproducible from the instance

pool and the pipeline alone, without recourse to any judgement exercised after the model outputs were seen. What the chapter does not claim is equally important and is consistent with the scope set out in Section 1.4: the results are a demonstration that the instrument discriminates between two models under a fixed protocol, not a general ranking of the models for professional deployment, nor a claim about their suitability for unsupervised use in practice. The implications of the comparison for the framework's standing, together with the limits of what two publicly accessible models under a single administration window can establish, are taken up in Chapter 6.

Chapter 6

Discussion

The preceding chapters developed a framework for evaluating Large Language Models on Project Management and Business Analysis processes, established its empirical warrant through two practitioner surveys, and applied it to a comparison of two contemporary models. This chapter interprets that result. It reads the validation and the execution against the questions the thesis set out to answer, interprets what the comparative evidence adds, draws out the implications for the framework and the practice it addresses, and states candidly the limits of what has been demonstrated. The discussion is deliberately even-handed: the framework's strengths are stated against the questions it was built to answer, and its limits against the objections a critical reader is entitled to raise.

6.1 The Findings in Light of the Research Questions

The thesis was organised around four research questions; the two surveys and the executed benchmark together supply the evidence that answers them, mapped in Table 6.1. The discussion interprets the mapping rather than restating it.

The first question concerned process selection. The first survey answers it not merely by ranking the seven macro-processes but by revealing the structure of professional opinion beneath the ranking: every macro-process is rated above the neutral midpoint, so none is dismissed, and the priority ordering is the resolution of two coherent viewpoints rather than a flat average. The domain-loyalty pattern, in which each community rates the processes it owns most highly, is the decisive evidence for the integrative choice, since a single-discipline framework would have inherited a systematic blind spot toward the other discipline's priorities. The selection is defensible not because consensus was uniform but because the framework accommodates a disagreement that is itself a feature of the professional landscape.

The second question concerned the criteria, and its two halves were addressed by different means. Automatability is a property of the design: each criterion is a deterministic rule comparing a structured output to a pre-specified ground truth over a closed input. Validity is established empirically: the second survey found every criterion endorsed, none significantly below threshold, and the decisive preference for multi-method scoring supports the two-criteria design. The conjunction is what matters, since automatability without validity yields a

reproducible benchmark that measures the wrong thing, and validity without automatability yields a sound instrument that cannot be applied at scale.

The third question concerned validation itself, and here the contribution is the protocol as much as its outcome. Filtering a framework on two distinct dimensions, relevance and measurement validity, with partially distinct cohorts, is a transferable method for grounding a design artefact in practitioner judgement; its principal signal is not any single mean but the convergence of independent lines of evidence on the same choices, which, together with the stability of ratings across roles and experience, licenses the claim to face and content validity.

The fourth question concerned what the validated instrument reveals in use, and the execution of Chapter 5 answers it directly. Administered to two models of differing capability over the full instance pool, the framework produced a clear and interpretable spread: an overall separation of twenty-six points, a clean ordering on eleven of twelve tasks, convergence on the single task whose structure leaves least room for capability, and a joint compression of both models on the hardest tasks. This is the behaviour of a working instrument rather than of a permissive metric, and, as Section 6.2 develops, the discrimination it exhibits is itself the first retrospective evidence bearing on the construct validity that the surveys could not establish.

Table 6.1: Research questions and the evidence that answers them.

Research question	Evidence
RQ1 — Which PM and BA processes are relevant to, and suitable for, the evaluation of model support, and how can a defensible subset be selected?	First survey: all seven macro-processes rated above the neutral midpoint, four at core relevance; the domain-loyalty pattern across roles vindicates the integrated design (Sections 4.3.2–4.3.3).
RQ2 — How can evaluation criteria be defined that are at once fully automatable and judged valid by domain experts?	Second survey: all twenty-four criteria validated, none significantly below threshold; the preference for multi-method scoring supports the two-criteria design; the criteria are deterministic rules over closed inputs (Sections 4.4.2–4.4.5).
RQ3 — Can the framework be validated through structured practitioner consensus?	The two-survey protocol was executed end to end; convergence of the quantitative and qualitative evidence, and stability across roles and experience, establish face and content validity (Sections 4.4.6–4.5).
RQ4 — When applied to contemporary models, what differences does the framework reveal, and does it produce a meaningful, reproducible spread?	Execution: two models scored over 998 instances under a fixed protocol yield a 26-point overall spread (0.807 vs 0.545), a clean ordering on 11 of 12 tasks, and an interpretable difficulty-sensitive profile (Sections 5.4–5.6).

6.2 Interpretation of the Benchmark Results

The execution contributes to this discussion in three ways that the task-level analysis of Chapter 5 does not, by design, draw out.

The first is evidential. A metric that separates systems of differing capability is, in doing so, shown to be measuring something real; the twenty-six-point spread is therefore the first retrospective construct-validity evidence for the instrument, complementing the face and content validity established in Chapter 4 and partially answering the limitation of Section 6.4.1. The evidence is strengthened, not weakened, by the shape of the spread: the instrument does not separate the models by a constant offset but compresses both on the hardest tasks and converges them on the most constrained one, which is what an instrument sensitive to task difficulty rather than to a fixed ranking should do.

The second concerns the role-aware reading that the validation motivated. The execution shows the stronger model's advantage to be markedly larger on the structured Project Management tasks than on the judgement-laden Business Analysis tasks, while the weaker model performs at almost the same level on both. The domain-loyalty pattern of the first survey thus finds a counterpart in the results: an aggregate score computed without regard to professional context would average over a real structure, in which the disciplines differ in how strongly they reward capability. The segmented interpretation that the validation recommended is therefore vindicated by the execution rather than merely anticipated by it.

The third is a caution the results make visible. Part of the weaker model's deficit on certain tasks is a failure of fidelity to the prescribed output structure rather than of domain reasoning, since outputs that did not conform to the specified form were scored as errors rather than repaired. For a benchmark of operational task performance this is the correct treatment, because answering in the specified form is part of the task; but a reader interpreting the comparison should hold instruction-following and domain reasoning apart, as the framework deliberately does not. The point bounds the strength of any claim drawn from the comparison and is taken up again among the limitations.

6.3 Implications

6.3.1 Methodological Implications

The most portable contribution of the thesis is methodological. The two-survey validation protocol is a template for any attempt to build a domain-specific evaluation instrument for generative models: relevance and measurement validity are separable concerns, and filtering them separately prevents the common conflation of “this process matters” with “this is how to score it”. The decision logic is similarly transferable: separating validation from selection, so that a criterion is first judged acceptable and only then, where alternatives exist, chosen as the operational metric, keeps two distinct questions from contaminating one another. The treatment of the threshold generalises too, since pairing a fixed cut-off with a tolerance band and a one-sample test against the reference value acknowledges sampling error honestly and prevents a difference of a few hundredths from deciding a criterion's fate; the single genuinely marginal case in the instrument was resolved by exactly this logic.

6.3.2 Implications for the Integrated Framework

The validation and the execution together reposition the framework from a proposal into a practitioner-endorsed instrument whose discriminating behaviour has been observed, which is the precondition for treating the scores it produces as meaningful rather than arbitrary. The domain-loyalty pattern, confirmed by the domain asymmetry of the results, implies that the framework is best understood as a measurement instrument whose output invites role-aware interpretation: the weight a community places on a process should inform how it reads a model's performance on that process. This is not a weakness of the instrument but a property of the domain it measures.

6.3.3 Implications for Practice

For practice, the framework addresses a concrete gap between the speed at which generative tools have been adopted in PM and BA work and the slower development of any disciplined means of assessing them. An organisation contemplating the use of a model for stakeholder identification, risk prioritisation, or requirements classification has, at present, little beyond anecdote on which to base the choice; a validated, fully automatable benchmark offers a reproducible alternative, since the same tasks, ground truth, and deterministic scoring can be applied to any model and re-applied as models evolve at negligible marginal cost. The value to practice is therefore not the one-time verdict of Chapter 5 but the durable instrument that produced it.

6.4 Limitations

The contributions above are bounded in ways a critical reading will identify, and the credibility of the work depends on stating those bounds plainly. Five limitations are principal; Table 6.2 summarises them and their disposition.

6.4.1 Face Validity Is Not Measurement Accuracy

The most consequential limitation is also the most easily misread. The two surveys establish that domain experts judge the framework's processes relevant and its criteria sensible; this is face and content validity, the appropriate standard for a design-stage artefact (Lawshe, 1975; Polit & Beck, 2006), not a demonstration that the metrics measure response quality with absolute accuracy, which requires the criterion or construct validity that accrues only when an instrument is applied and its scores related to an external standard (Cronbach & Meehl, 1955). The execution of Chapter 5 furnishes a first instalment of such evidence, since a metric that discriminates between systems of differing capability is thereby shown to measure something real, but it is an instalment and not a settlement: the absolute calibration of the metrics against an external quality standard remains to be established. What bounds the limitation without dissolving it is that the criteria are deterministic functions of a structured output and a fixed ground truth, so their behaviour is transparent and reproducible even where their absolute calibration is not yet fixed.

6.4.2 Sampling

Both surveys recruited through a professional network, so the samples are samples of convenience, subject to self-selection and to the concentration of that network; their sizes, forty and thirty-three, do not support population-level inference, which is why subgroup comparisons were reported as exploratory. Three considerations mitigate rather than remove the limitation: expert-panel content validity is conventionally established with panels of comparable or smaller size (Lynn, 1986); the endorsement is stable across roles and experience bands, which argues against its being an artefact of sample composition; and the two surveys, on partially overlapping cohorts, converge on the same choices. The work is accordingly presented as a preliminary expert validation, with the boundary declared rather than obscured.

6.4.3 The Closed-Candidate-List Trade-off

Several tasks present the model with a closed list and ask it to select, classify, or rank rather than generate freely. As a respondent observed, this departs from practice, in which the practitioner must generate the candidates as well as evaluate them, so the closed list tests a filtering ability rather than the full generative task. The observation identifies a genuine trade-off: the pattern is a deliberate device that buys objective, reproducible, human-free scoring at the cost of some ecological fidelity. The cost is partly recovered by the richer prompt configurations, and the design of open-generation variants that preserve automatable scoring is identified as a direction for a subsequent iteration.

6.4.4 Choice and Coverage of the Scoring Techniques and Criteria

The scoring techniques underlying several criteria are drawn from information retrieval rather than from the PM or BA canon, and treating Likert ratings parametrically for aggregation rests on an established but not uncontested convention (Norman, 2010). Neither is a hidden defect: the operationalised tasks reduce to selection, classification, and ranking against a ground truth, and the techniques adopted are the established, reproducible means of scoring exactly those response types. The two-criteria architecture is likewise a deliberate parsimony, endorsed by respondents, though some tasks might be served by a third criterion. Two dimensions the present criteria do not capture, sensitivity to prompt phrasing and consistency across repeated runs, were flagged by respondents as candidate meta-criteria and are carried forward to a future iteration.

6.4.5 The Scope of the Execution

The execution itself is bounded in ways that limit what its comparison can establish. It evaluates two models from two providers, both through their standard consumer interfaces, over a single administration window; it is therefore a demonstration that the instrument discriminates under a fixed protocol, not a general or current ranking of models for professional deployment. Two further bounds follow from the administration. The weaker model required a declared second pass to complete two of the longer blocks, an asymmetry reported transparently but one that means the two models were not administered under byte-identical conditions throughout. And, as Section 6.2

noted, the framework does not separate a deficit of instruction-following from a deficit of domain reasoning, so a part of the observed gap on some tasks reflects output-form fidelity rather than professional competence. None of these bounds undermines the demonstration, but each constrains the generality of any claim drawn from it.

Table 6.2: Principal limitations, their mitigation, and their disposition.

Limitation	Mitigation	Disposition
Face and content validity, not absolute measurement accuracy	Appropriate, declared design-stage standard; deterministic, reproducible metrics; the execution supplies first construct-validity evidence	Partly addressed by the execution; full calibration on continued use
Convenience sampling and modest sample sizes	Consistent with expert-panel practice; stability across roles and experience; convergence of two surveys	Accepted and declared
Closed-candidate-list design limits ecological validity	Deliberate trade-off securing objective, automatable scoring; richer prompt configurations partially recover realism	Future iteration
Choice and coverage of scoring techniques and criteria	Techniques chosen for reproducibility over the operationalised task types; parsimonious two-criteria design endorsed by respondents	Future iteration
Scope of the execution: two publicly accessible models, one window; instruction-following not separated from reasoning	Demonstration of discrimination under a fixed protocol, transparently reported; second pass declared	Extended in future work

6.5 Concluding Remarks

Taken together, the discussion supports a measured claim. The framework answers the questions the thesis posed: it selects a defensible set of processes whose relevance is grounded in practitioner judgement, it defines evaluation criteria that are at once fully automatable and endorsed as valid, it does so through a validation protocol whose convergence is its principal warrant, and, once executed, it discriminates between models of differing capability with an interpretable, reproducible spread. The claim is measured because the validation it rests on is one of face and content validity, established on samples of convenience and on an instrument whose closed-list design and reproducibility-driven scoring trade ecological fidelity for measurement validity, and because the execution that complements it spans two publicly accessible models under a single protocol. These bounds do not undermine the contribution; they locate it. The thesis delivers a practitioner-validated, fully automatable instrument for evaluating model support of Project Management and Business Analysis work, and the comparative evidence of Chapter 5 shows that instrument doing what it was built to do. The concluding chapter draws the contributions together and

situates them within the broader trajectory of domain-specific evaluation.

Chapter 7

Conclusion

This thesis set out to address a specific asymmetry: while Large Language Models are increasingly used to support Project Management and Business Analysis work, the means by which their performance on that work is assessed have remained fragmented, task-bound, and dependent on scoring procedures that resist reproduction. The response developed across the preceding chapters is a single methodological instrument: a framework that selects a defensible set of professional tasks, operationalises each as a benchmark item with a closed input and a structured output, scores each through fully automated criteria, rests the whole on the judgement of practitioners from both communities, and is carried through to a concrete comparison of contemporary models. The discussion weighed what the instrument does and does not establish; the purpose here is to draw the contributions together, to situate them within the wider movement toward domain-specific evaluation of generative models, and to set out the work that follows from them.

7.1 Summary of Contributions

The thesis makes five contributions, each answering to one of the research gaps identified at the outset and together constituting the deliverable the work was undertaken to produce.

The first is an integrated evaluation framework spanning Project Management and Business Analysis within a single taxonomy. Where the existing literature offers bibliometric mappings and isolated benchmarks each confined to one task, the framework positions twelve operational tasks on the canonical grids of the PMBOK Guide and the PMI Guide to Business Analysis, deliberately balanced across the two disciplines. The integration is not a juxtaposition of two benchmarks but a coherent instrument in which the shared treatment of stakeholders and requirements and the divergence between a delivery orientation and an analysis orientation are both represented; the validation showed this division to be a pattern in how practitioners themselves rate relevance, which is the strongest available warrant for an integrated rather than a siloed design.

The second is a system of twenty-four evaluation criteria, two per task, that are fully automatable without sacrificing professional meaning. Each computes a score from a structured output and an ex-ante ground truth through elementary arithmetic alone, with no semantic matching and no expert rubric at the moment of scoring, and the

criteria are organised into three families of methodological trade-offs so that the system is a set of framework-level patterns rather than a list of particularities. Reaching full automation across task types as varied as numerical estimation, set selection, classification, ranking, and relational-graph construction is the methodological core of the work, and the closed-candidate-list pattern is the device that makes it possible.

The third is a two-survey validation methodology that supplies the empirical evidence on which the framework's defensibility rests. The first survey filtered macro-processes for professional relevance; the second submitted the criteria to practitioners for a judgement of methodological validity, criterion by criterion. The result is a validation whose principal warrant is the convergence of independent lines of evidence rather than the weight of any single test, placing the framework on a footing of face and content validity established through structured expert consensus.

The fourth is the adaptation of the holistic, multi-metric benchmarking paradigm to a domain it had not previously reached. The foundational literature established the scenario-by-criterion method as the standard for evaluating language models in general, but none of its instruments contains scenarios drawn from Project Management or Business Analysis, and the single discipline-anchored benchmark tests knowledge recognition through multiple-choice questions rather than the capacity to produce professional artefacts. The framework inherits the methodological commitments of holistic evaluation and specialises them to the operational tasks that define practice in these fields.

The fifth is the execution of the validated instrument against contemporary models. Administered to two general-purpose models over the full instance pool under a single fixed protocol, and scored entirely by the deterministic pipeline using the official metric of each task, the framework produced the first comparative profile of contemporary LLMs across an integrated set of Project Management and Business Analysis tasks (Chapter 5). In doing so it demonstrated that the instrument discriminates between models of differing capability with an interpretable, reproducible spread, converting the framework from a validated proposal into an instrument observed in use.

7.2 Answer to the Research Questions

The five contributions resolve the four research questions in turn, and in the same order of dependency in which the questions were posed. The selection of processes and their operationalisation into benchmarkable tasks, the matter of the first question, is answered by the first contribution together with the relevance evidence of the first survey: practitioners endorse the macro-processes, and the framework translates them into twelve tasks through operational decomposition and dual framing. The identification of criteria at once automatable and professionally valid, the matter of the second question, is answered by the second and third contributions jointly: the criteria are automatable by design and shown to be valid by the second survey. The sufficiency of practitioner consultation as the empirical basis for a defensible instrument, the matter of the third question, is answered by the third contribution, with the qualification stated plainly in the discussion that the warrant obtained is one of expert-consensus validity rather than of demonstrated measurement accuracy against an external standard.

The fourth question, what the validated instrument reveals when applied to contemporary models and whether it produces a meaningful and reproducible spread, is answered by the fifth contribution and the execution of

Chapter 5. The answer is affirmative: the instrument separates two models of differing capability by twenty-six points overall, orders them cleanly on eleven of twelve tasks, and does so with a profile sensitive to task difficulty rather than fixed by a constant offset. That discrimination is itself a first instalment of the construct-validity evidence the surveys could not provide, with the qualification, equally stated in the discussion, that the result is a demonstration of the instrument under a fixed protocol and not a general ranking of the models for professional deployment. The four answers are mutually dependent in the way the questions were: the tasks supply the ground on which the criteria operate, the criteria supply the object the validation judges, the validation supplies the evidence on which tasks and criteria can be claimed defensible, and the execution supplies the demonstration that the validated instrument behaves as one in use.

7.3 Toward Domain-Specific Evaluation of Generative Models

The contribution is best understood against a broader movement. As generative models are absorbed into professional work, the question of how to evaluate them is migrating from general capability, measured on broad academic and reasoning benchmarks, toward fitness for specific professional purposes, measured on the operational tasks a discipline actually performs. General benchmarks answer whether a model is capable in the abstract; they do not answer whether it produces a sound work breakdown structure, identifies the right stakeholders from a project charter, or extracts the explicitly stated needs from an elicitation transcript. Domain-specific evaluation answers the second kind of question, and it requires exactly what this thesis has sought to supply: a taxonomy that fixes which tasks count, an operationalisation that makes outputs comparable, scoring that can be reproduced, and validation that grounds the choices in the judgement of the professionals whose work is at stake.

The framework is a contribution to this movement rather than its completion. Its value lies as much in the design pattern it embodies as in the twelve tasks it currently covers, and the execution of Chapter 5 shows that the pattern, once instantiated, yields evidence of the kind domain-specific evaluation needs. This locates the durable value of the work where the most recent benchmarking canon locates it, in the questions, the candidate answers, and the validated evaluation criteria rather than in any particular pair of scores, which are properties of the models under test at a moment in time and are expected to change as the technology evolves (Center for AI Safety et al., 2026). The instrument is, in this sense, an opening onto the domain-specific evaluation of generative models in these professions rather than a closed account of it.

7.4 Future Work

The most direct continuation is the extension of the framework to the Project Management and Business Analysis processes it does not yet cover. Its value lies as much in the design pattern as in the twelve tasks, and the same template, a taxonomy coordinate, a closed-input structured-output question, a candidate list where applicable, and two criteria drawn from the three families, can carry the framework to the remaining Performance Domains of the

PMBOK Guide and Knowledge Areas of the PMI Guide to Business Analysis. It can in particular accommodate the process areas that practitioners themselves named as missing in the surveys, among them quality management, procurement, project closing and lessons learned, status and performance reporting, and solution evaluation. Several of these produce free-form rather than closed outputs, so their inclusion is bound up with the second direction below, and the method by which the present twelve tasks were validated applies unchanged to whatever tasks are added.

A second direction concerns the boundary the closed-candidate-list pattern draws. The pattern secures full automation by converting open-ended production into selection from a fixed inventory, which is the trade of ecological fidelity for measurement validity that the discussion identified as a limitation. A complementary line of work would develop open-generation tasks, in which a model produces a project artefact in free form, paired with evaluation methods suited to that setting; such tasks would test the capacity to generate rather than to select, and would position closed-list and open-generation evaluation as complementary halves of a fuller account of model support for this work.

A third direction concerns two meta-criteria the validation surfaced but that lie beyond per-task scoring: a model's sensitivity to how a prompt is phrased, and its consistency across repeated runs of the same prompt. The framework already makes the first measurable, since its prompt configurations hold the item fixed while varying the framing, so that the variation in score is itself a measure of prompt sensitivity; the second requires repeated execution of identical prompts and the quantification of the resulting dispersion. Developing both into first-class evaluation dimensions would extend the framework from a measure of how well a model performs to a measure of how dependably it does so.

Finally, because the instrument is reproducible at negligible marginal cost, the comparison of Chapter 5 can be re-run across a broader and continually evolving set of models as the technology changes, extending the demonstration without altering the instrument. Together with the further tasks and knowledge areas the framework's template can accommodate, these directions describe a programme of which the present thesis is the first foundational instalment.

References

- Almeida, C. et al. (2025). “From Elicitation Interviews to Software Requirements: Evaluating LLM Performance in Requirement Generation”. *Proceedings of the Workshop on Requirements Engineering (WER)*.
- Anderson, L. W. & Krathwohl, D. R. (2001). *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives*. New York: Longman.
- Barcaui, A. & Monat, A. (2023). “Who is better in project planning? Generative artificial intelligence or project managers?” *Project Leadership and Society*. 10.1016/j.plas.2023.100101.
- Beg, A. et al. (2025). *A Short Survey on Formalising Software Requirements using Large Language Models*. 10.48550/arXiv.2506.11874. arXiv: 2506.11874 [cs.SE].
- Carpenter, J. et al. (2024). *Leveraging Large Language Models for Predicting Cost and Duration in Software Engineering Projects*. 10.48550/arXiv.2409.09617. arXiv: 2409.09617 [cs.SE].
- Center for AI Safety et al. (2026). “A benchmark of expert-level academic questions to assess AI capabilities”. *Nature*, 649:8099, 1139–1146. 10.1038/s41586-025-09962-4.
- Chen, D.-L. et al. (2025). “The Design and Implementation of an Educational Chatbot with Personalized Adaptive Learning Features for Project Management Training”. *Technology, Knowledge and Learning*. 10.1007/s10758-024-09807-5.
- Cronbach, L. J. & Meehl, P. E. (1955). “Construct validity in psychological tests”. *Psychological Bulletin*, 52:4, 281–302. 10.1037/h0040957.
- Ellsel, C. & Stark, R. (2025). “Advancing Requirements Engineering with Large Language Models”. *Procedia CIRP*. 10.1016/j.procir.2025.08.120.
- Erfani, A. & Khanjar, H. (2025). “Large Language Models for Construction Risk Classification: A Comparative Study”. *Buildings*. 10.3390/buildings15183379.

- Gu, W. et al. (2025). “Large language model-empowered dynamic scheduling for intelligent hybrid flow shop using multi-agent deep reinforcement learning”. *Advanced Engineering Informatics*. 10.1016/j.aei.2025.104294.
- Hendrycks, D. et al. (2021). “Measuring Massive Multitask Language Understanding”. *Proceedings of the International Conference on Learning Representations (ICLR)*. arXiv: 2009.03300.
- International Organization for Standardization (2018). *ISO/IEC/IEEE 29148:2018 – Systems and software engineering – Life cycle processes – Requirements engineering*. Tech. rep. Geneva: International Organization for Standardization.
- Järvelin, K. & Kekäläinen, J. (2002). “Cumulated gain-based evaluation of IR techniques”. *ACM Transactions on Information Systems*, 20:4, 422–446. 10.1145/582415.582418.
- Karnouskos, S. (2024). “The Relevance of Large Language Models for Project Management”. *IEEE Open Journal of the Industrial Electronics Society*, 5, 758–768. 10.1109/OJIES.2024.3412222.
- Khalil, M. et al. (2025). “Mapping the AI Landscape in Project Management Context: A Systematic Literature Review”. *Systems*. 10.3390/systems13100913.
- Lawshe, C. H. (1975). “A quantitative approach to content validity”. *Personnel Psychology*, 28:4, 563–575. 10.1111/j.1744-6570.1975.tb01393.x.
- Li, M. et al. (2026). “A4PS: Agentic AI-assisted advanced planning and scheduling with large language models for smart manufacturing”. *Journal of Manufacturing Systems*.
- Liang, P. et al. (2023). “Holistic Evaluation of Language Models”. *Annals of the New York Academy of Sciences*, 1525:1, 140–146. 10.1111/nyas.15007.
- Liu, J. et al. (2026). “CART: A traceable zero-shot planning framework for large language models with adaptive replanning”. *Knowledge-Based Systems*. 10.1016/j.knosys.2025.115189.
- Lubos, S. et al. (2024). *Leveraging LLMs for the Quality Assurance of Software Requirements*. 10.48550/arXiv.2408.10886. arXiv: 2408.10886 [cs.SE].
- Lynn, M. R. (1986). “Determination and quantification of content validity”. *Nursing Research*, 35:6, 382–385. 10.1097/00006199-198611000-00017.
- Martin, H. et al. (2025). “Exploring Large Language Model AI tools in Construction Project Risk Assessment: ChatGPT Limitations in Risk Identification, Mitigation Strategies, and User Experience”. *Journal of Construction Engineering and Management*. 10.1061/JCEMD4.COENG-16658.

- Matera, M. et al. (2026). “Leveraging LLMs for Conversational Data Access: A Human-Centred Perspective”. *Lecture Notes in Computer Science*. 10.1007/978-3-031-97207-2_3.
- Mircea, M. et al. (2026). “Supporting Stakeholder Requirements Expression with LLM Revisions: An Empirical Evaluation”. *Proceedings of the International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ)*. 10.48550/arXiv.2601.16699.
- Norman, G. (2010). “Likert scales, levels of measurement and the “laws” of statistics”. *Advances in Health Sciences Education*, 15:5, 625–632. 10.1007/s10459-010-9222-y.
- Polit, D. F. & Beck, C. T. (2006). “The content validity index: Are you sure you know what’s being reported? Critique and recommendations”. *Research in Nursing & Health*, 29:5, 489–497. 10.1002/nur.20147.
- Project Management Institute (2017). *The PMI Guide to Business Analysis*. Newtown Square, PA: Project Management Institute.
- (2021). *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 7th. Newtown Square, PA: Project Management Institute.
- Radliński, Ł. & Swacha, J. (2025). “Large Language Models for Early-Stage Software Project Estimation: A Systematic Mapping Study”. *Applied Sciences*, 15:24, 13099. 10.3390/app152413099.
- Saleem, S. et al. (2026). “ReqNet: an LLM-driven computational framework for automated requirements extraction from unstructured documents”. *Complex & Intelligent Systems*. 10.1007/s40747-025-02143-w.
- Sami, M. A. et al. (2026). “Bridging Humans and LLMs: Investigating Human-AI Collaboration in Multi-agent Requirements Analysis for Organizational AI Adoption”. *e-Informatica Software Engineering Journal*. 10.37190/e-Inf260103.
- Sarim, M. et al. (2025). “Generating reliable software project task flows using large language models through prompt engineering and robust evaluation”. *Scientific Reports*. 10.1038/s41598-025-19170-9.
- Shahriary, A. et al. (2025). “Assessing Large Language Models as Agile Scrum Masters: A Comparative Study of Project Planning Efficiency”. *Proceedings of the International Conference on Web Research (ICWR)*. IEEE. 10.1109/ICWR65219.2025.11006172.
- Shaikh, S. H. (2026). “LLM-Based Multi-agent Systems: Frameworks, Evaluation, Open Challenges, and Research Frontiers”. *Communications in Computer and Information Science*. 10.1007/978-3-032-15632-7_9.

- Snell, Q. et al. (2025). “Assessing Large Language Models in Building a Structured Dataset From AskDocs Subreddit Data: Methodological Study”. *Journal of Medical Internet Research*. 10.2196/74094.
- Spearman, C. (1904). “The Proof and Measurement of Association between Two Things”. *The American Journal of Psychology*, 15:1, 72–101. 10.2307/1412159.
- Srivastava, A. et al. (2023). “Beyond the Imitation Game: Quantifying and Extrapolating the Capabilities of Language Models”. *Transactions on Machine Learning Research*. arXiv: 2206.04615.
- Van Rijsbergen, C. J. (1979). *Information Retrieval*. 2nd. London: Butterworths.
- Wang, Y. et al. (2025). “Multi-agent large language models as evolutionary optimizers for scheduling optimization”. *Computers and Industrial Engineering*. 10.1016/j.cie.2025.111197.
- Wysocki, W. & Ochodek, M. (2026). “Leveraging LLM-based data augmentation for automatic classification of recurring tasks in software development projects”. *Journal of Systems and Software*. 10.1016/j.jss.2025.112641.
- Zadenoori, M. A. et al. (2025). *Large Language Models (LLMs) for Requirements Engineering (RE): A Systematic Literature Review*. 10.48550/arXiv.2509.11446. arXiv: 2509.11446 [cs.SE].
- Zhang, M. et al. (2026). “Automating construction contract question answering using large language model and fine-tuning”. *Expert Systems with Applications*. 10.1016/j.eswa.2025.129493.

Appendix A

First Survey Instrument and Synthesis

The first practitioner survey established the professional relevance of the macro-processes from which the twelve benchmark tasks are derived. It was administered as an anonymous online questionnaire in English, comprising twelve questions, three of them optional, with a completion time of approximately seven minutes. The instrument was organised into seven parts: a respondent profile covering professional role, years of experience, and familiarity with generative-AI tools; a process-relevance block scored on a five-point Likert matrix over the seven macro-processes; a task-formulation block assessing the quality of the candidate task statements; two criteria-appropriateness blocks covering the general and the Business-Analysis-specific evaluation criteria respectively; an evaluation-method preference question; and a closing free-text section inviting respondents to name processes outside the surveyed set and to comment on the proposed approach.

The relevance block and the open-response items supply the empirical input analysed in Chapter 4. The survey was completed by forty respondents, balanced across the two target communities and their adjacent roles, with Project Managers the largest single group, followed by Business Analysts and the Project Management Office and a smaller number of Product Managers, Consultants, and one respondent from an adjacent operations role; the role and experience composition is reported in Table 4.1 of Chapter 4 and supports the role-segmented analyses presented there. The full item wording, the response distributions for each Likert block, the role cross-tabulations, and the coded open responses are retained in the survey synthesis workbook that accompanies the thesis materials.

Appendix B

Automated Scoring Pipeline

This appendix documents the automated scoring pipeline that operationalises the framework, described in prose in Section 3.7. The pipeline is the concrete, reproducible instrument of the thesis: it implements the official metric of every task as a deterministic function of a structured output and an ex-ante ground truth, assembles prompts under the configurations described in Section 3.7, and aggregates per-instance scores into per-task estimates separable by task, configuration, and model. No step involves semantic matching or human judgement at scoring time. The full source file accompanies the thesis materials; this appendix reproduces its architecture and representative scoring functions.

The pipeline is organised as a single pass from the item bank to the aggregated scores:

item bank $\rightarrow$ prompt assembly $\rightarrow$ model query $\rightarrow$ structured-output parsing per block $\rightarrow$ deterministic scoring (24 criteria) $\rightarrow$ aggregation (task / configuration / model).

The model-query step is a thin, model-agnostic interface: the scoring layer is validated independently of any model run through built-in self-tests that reproduce the worked examples documented for each evaluation criterion, so that the correctness of the implementation can be checked without network access. The official metric retained for each task, fixed by the second survey, is the one reported in Table 4.5; the pipeline records these assignments in a registry and applies the corresponding function to each task block.

Every criterion reduces to one of four families of elementary-arithmetic measures. The balanced selection-accuracy measure scores set selection as $B = 2c/(2c + w + m)$, where c , w , and m are the counts of correctly selected, wrongly selected, and missed items, with the wrong-selection term carried as a real number so that specific classes of false positive can be down-weighted. The rank-agreement measure scores an ordering as $R = 1 - \sum |r_{\text{pred}} - r_{\text{true}}|/D$, with D the maximum attainable displacement. The top-positions-weighted measure scores a ranked selection as $W = \sum \max(0, \text{weight} - \text{displacement})/W_{\text{max}}$, penalising errors near the top of the list more heavily. The tolerance-band and ratio measures score the single-number cost task. The information-retrieval names of these families (the F1-score, Spearman’s rank correlation, and the normalised Discounted Cumulative Gain) are stated only in Chapter 3; the implementations are the elementary-arithmetic versions that constitute the official benchmark metrics.

Two representative functions illustrate the scoring layer. The first is the balanced selection-accuracy measure

used by the majority of the criteria; the second is the official metric of the cost-estimation task and its retained alternative.

```
def balanced_selection(correct, wrong, missed):
    """Balanced selection-accuracy measure  $B = 2c / (2c + w + m)$ .
    Returns 1.0 for the empty-vs-empty degenerate case; `wrong` is a
    float so callers may down-weight specific false-positive classes."""
    denom = 2 * correct + wrong + missed
    if denom == 0:
        return 1.0
    return clamp01((2 * correct) / denom)

def score_pm_ext_01_c1(resp, gt):
    """Official metric: proportional accuracy on the single cost figure."""
    est, true = resp.get("cost_eur"), gt["cost_eur"]
    if est is None or true == 0:
        return 0.0
    return clamp01(1.0 - abs(float(est) - true) / abs(true))

def score_pm_ext_01_c2(resp, gt):
    """Retained alternative: discrete tolerance bands on relative error."""
    est, true = resp.get("cost_eur"), gt["cost_eur"]
    if est is None or true == 0:
        return 0.0
    err = abs(float(est) - true) / abs(true)
    if err <= 0.10: return 1.00
    if err <= 0.25: return 0.75
    if err <= 0.50: return 0.50
    return 0.0
```

The remaining twenty-one scoring functions follow the same pattern, each consuming a parsed structured output and the ex-ante ground truth of the instance and returning a score on the unit interval. An output that cannot be parsed against the structure prescribed for its task is scored as null performance, consistent with the treatment of non-conforming outputs described in Section 5.3.

Appendix C

Second Survey Instrument

The second practitioner survey validated the evaluation criteria of the framework. It was administered as an anonymous online questionnaire in English, with a completion time of approximately ten to fifteen minutes, and was completed by thirty-three respondents whose composition is reported in Table 4.3 of Chapter 4. The instrument was organised in two parts. The first part profiled the respondent through four questions covering professional role, years of experience, familiarity with generative-AI tools in a work setting, and whether the respondent had also taken part in the first survey. The second part presented the twelve tasks one block at a time; for each block the respondent was shown the process name and a brief description, the concrete question that would be submitted to a model together with an input example, and one or two candidate evaluation criteria, each described in plain language with a worked numerical example. For every criterion the respondent answered a single question on a one-to-five scale: whether it makes sense to evaluate the accuracy of the model's response on that process through that criterion. Each block closed with an optional free-text field inviting the respondent to indicate whether a low rating concerned the question, the criterion, or both. The criteria were presented through their operational behaviour alone; the information-retrieval names of the underlying measures were withheld from respondents so that the rating would reflect the behaviour of each criterion rather than familiarity with any statistical notation.

One block is reproduced here to illustrate the instrument; the remaining eleven follow the same structure, and the full questionnaire accompanies the thesis materials.

Block 1 / 12 — Estimate Costs (PMBOK, Finance × Planning)

Process description. Develop an approximation of the monetary resources needed to complete the project activities.

Question submitted to the model. Based on the attached project description, provide a cost estimate for the project in euros, expressed as a single number. Input: a one-to-two-page document describing scope, deliverables, estimated duration, and team composition. Expected output: a number (e.g. €127,500).

Criterion 1 — proportional accuracy. The output is compared to the true project cost; the score decreases smoothly with distance from the true value, so that an exact match scores 1.0 and a ten-per-cent deviation scores 0.90. *Example:* true cost €100,000, estimate €95,000 (five per cent

off), score 0.95. *Rate 1–5: does it make sense to evaluate accuracy on this process through this criterion?*

Criterion 2 — discrete tolerance bands. The estimate scores 1.0 within $\pm 10\%$ of the true value, 0.75 within $\pm 20\%$, 0.50 within $\pm 30\%$, and 0 beyond, reflecting the rough, budget, and definitive estimate categories used in practice. *Example:* true cost €100,000, estimate €115,000 (fifteen per cent off, within the $\pm 20\%$ band), score 0.75. *Rate 1–5: does it make sense to evaluate accuracy on this process through this criterion?*

Optional. If you rated a criterion low, indicate whether the issue lies in the question, in the criterion, in both, or elsewhere.

Appendix D

Benchmark Item-Bank

The execution reported in Chapter 5 was conducted over an item bank of nine hundred and ninety-eight instances generated by deterministic, seeded parametric templates, one per task. Each instance carries an identifier, its task block, the prompt input, the ex-ante ground truth, and generation metadata; the ground truth of every instance is computed by the same rules that produce the instance, so that the reference is internal to the framework and depends on no external annotator, and the entire bank is auditable and exactly reproducible from its templates and per-block seeds.

The bank was assembled in three generation waves, summarized in Table D.1. The waves group the tasks by output structure rather than by discipline: the first covers the numeric and structured-table PM tasks, the second a mixed set of selection and classification tasks across both disciplines, and the third the three high-volume ranked- and closed-selection tasks that carry the largest instance counts under the relevance-proportional allocation of Section 3.2.6. The instances per-task are those reported in Table 5.1 and visualized in Figure 3.1; the full item bank, with prompt input and ground truth of every instance, accompanies the thesis materials, and one representative instance per task is reproduced verbatim in Section 3.3.2.

Table D.1: Composition of the benchmark item-bank across the three generation waves.

Wave	Task blocks	Instances
1	Estimate Costs, Sequence Activities, Develop Schedule, Plan Scope	216
2	Develop PM Plan, Prioritize Requirements, Define Requirements, Conduct Stakeholder Analysis, Identify Stakeholders (PM)	409
3	Identify Stakeholders (BA), Conduct Elicitation, Identify Risks	373
Total		998

