Appendix

Appendix A – Interview Guidelines

Interview-Round 1:

Introductory Questions

- Q1: What is your professional background and current role?

Current Project Risk Management Practice

- Q2: How do you currently identify and assess relevant risks in projects?
- Q3: What are the main challenges you experience in project risk management?
- Q4: Which tools or software systems do you currently use for project risk management?

AI in Project Risk Management

- Q5: Have you used AI-based tools in project risk management or related project management tasks?
- Q6: What concerns do you have regarding the use of AI in project risk management?

Potential Use of an LLM-Based Information Extraction System

- Q7: To what extent could it be useful if an application automatically extracted and assessed relevant project risk indicators from project-related documents and communication?
- Q8: Which specific functions would you expect from such an application?

Socio-Technical Requirements

- Q9: What non-technical requirements would such a tool need to fulfil?
- Q10: Under which conditions would you consider such a tool useful and practically applicable in your organization?

Closing Questions

- Q11: Do you have any additional remarks or feedback on the research project?
- Q12: Do you know further experts or practitioners who would be relevant to interview?

Interview-Round 2:

Introductory Questions

- Q1: What is your professional background and current role?

Current Project Risk Management Practice

- Q2: How do you currently identify and assess risks in projects?
- Q3: Where do early risk signals usually appear in your projects?
- Q4: Which types of risks are easiest to identify, and which are more difficult?

AI in Project Risk Management

- Q5: How do you currently use AI or LLM-based tools in project-related work?
- Q6: What benefits and concerns do you see in using AI for project risk management?

Evaluation of the Architecture

Brief introduction of the architecture

- Q7: What is your overall impression of the concept?
- Q8: Which data sources should such a system realistically ingest?
- Q9: How do you assess the idea of domain-specialized agents for different risk types?
- Q10: How important is the interpretation of informal or ambiguous communication?

Evaluation of the Prototype

Brief demonstration of the prototype:

- Q11: What is your overall impression of the prototype?
- Q12: Does the prototype provide a useful overview of project risk indicators?
- Q13: How frequently should the system surface risk indicators?
- Q14: Does the prototype provide sufficient verifiability and traceability?
- Q15: How do you assess the configuration and feedback mechanisms?

Practical Viability and Closing Questions

- Q16 For which types of projects would the system be most useful?
- Q17: What would be the main cost-benefit argument for adopting such a system?

- Q18: What would be the main barriers to adoption?
- Q19: If you had three wishes to improve the system, what would you change?
- Q20: Is there anything else you would like to add?

Appendix B – LLM Parameters and Prompt Templates

Parameter	Specification
Specialized Risk Agent (example: Technical)	
Model (tier)	Gemini 3 Flash
Reasoning mode	Off
Temperature	0.1
Top-p / Top-k	0.95 / 40
Max output tokens	2,048
Response format	JSON (schema-constrained)
System prompt template	**System prompt:** You are a Specialized Risk Agent for the domain {domain_name}. Your task is to analyze a single project document and identify at most one concrete risk indicator that belongs to your assigned domain. Project grounding: Use the project context only to interpret project-specific names, organizations, systems, milestones, sites, and terminology. Do not copy risks from the project context and do not invent new facts from background knowledge. Risk definition: A risk indicator is an explicit, text-grounded signal that something relevant to the project is unresolved, missing, delayed, blocked, unclear, disputed, failing, degraded, or already causing negative impact. Routine planning, ordinary action items, positive status updates, and neutral references to project work are not risk indicators unless the document states a blocker, lateness, uncertainty, missing prerequisite, disagreement, or consequence. Domain discipline: Extract only risks that fall within {domain_name}. If the document contains a risk in another domain, return risk_found=false and explain that the signal is outside your domain. If the document is out of project scope, return risk_found=false. Evidence rule: If risk_found=true, provide a verbatim evidence quote from the document. The evidence quote must directly support the description. The description must stay close to the quote and must not add unmentioned causes, impacts, dates, owners, or mitigation status. If no direct quote supports the finding, return risk_found=false. Severity and likelihood: Assign severity and likelihood conservatively. Use High only when the document indicates a critical milestone, compliance requirement, major delivery failure, substantial operational impact, or similarly severe exposure. Use Medium for material but bounded exposure. Use Low for localized, early, or weakly supported exposure. Likelihood should reflect the strength and directness of textual evidence, not general plausibility. **Output schema:** Return exactly one JSON object with the following fields: risk_found (boolean), domain (string), description (string or null), severity (High, Medium, Low, or null), likelihood (High, Medium, Low, or null), evidence_quote (string or null), signal_category (string or null), no_risk_reason (string or null). Return JSON only.
Orchestrator Agent	
Model (tier)	Gemini 3 Flash
Reasoning mode	Off
Temperature	0.2
Top-p / Top-k	0.95 / 40
Max output tokens	4,096
Response format	JSON (schema-constrained)
System prompt template	**System prompt:** You are the Orchestrator in a multi-agent project risk detection system. You receive one project document, the project context, the active risk domains, and the available Specialized Risk Agents. Your task is to assess whether the document is project-relevant and potentially risk-relevant, and to route it to the appropriate domain-specialized agent(s). You do not extract final risk indicators and you do not create the final risk list; extraction is performed by the Specialized Risk Agents and consolidation by the Aggregator Agent. Routing objective: Decide which Specialized Risk Agent(s) should analyze the document. Use the project context only to interpret project-specific names, organizations, systems, milestones, sites, and terminology. Do not infer risks from the project context alone and do not invent facts from background knowledge. Project-scope check: First determine whether the document belongs to the project scope. If the document concerns another project, a private matter, general organizational

	communication, or a source unrelated to the configured project, set project_relevant=false and do not route it to any Specialized Risk Agent. Risk-relevance check: If the document is project-relevant, assess whether it contains language that may indicate an unresolved problem, gap, delay, blocker, uncertainty, dispute, failure, degradation, missing prerequisite, or negative impact. Routine planning, ordinary action items, positive status updates, and neutral references to project work are not sufficient unless they contain such a potential signal. Domain routing: Select all and only those risk domains for which the document contains plausible textual cues. Multi-label routing is allowed when the same document contains cues for several domains, for example technical and financial risks. Do not route a document to a domain merely because the project context mentions that domain. If the document is project-relevant but contains no plausible risk-relevant cue, set risk_relevant=false and do not route it to any Specialized Risk Agent. Evidence for routing: For each selected domain, provide a short routing rationale and a brief verbatim cue from the document that motivated the routing decision. The cue does not need to prove that a risk exists; it only needs to justify why a Specialized Risk Agent should inspect the document. If no textual cue supports a domain, do not select that domain. Conservative routing: Prefer routing when the document contains a weak but plausible risk cue, because the Specialized Risk Agents perform the detailed extraction. However, avoid broad over-routing to unrelated domains, as this increases false positives and unnecessary processing. **Output schema**: Return exactly one JSON object with the following fields: project_relevant (boolean), risk_relevant (boolean), selected_agents (array of strings), routing_labels (array of strings), routing_rationale (object mapping each selected agent to a concise rationale), evidence_cues (object mapping each selected agent to a verbatim cue or null), confidence (High, Medium, Low), no_route_reason (string or null).
Aggregator Agent	
Model (tier)	Gemini 3 Pro
Reasoning mode	On (high)
Temperature	0.2
Top-p / Top-k	0.95 / 40
Max output tokens	4,096
Response format	JSON (schema-constrained)
System prompt template	**System prompt:** You are the Aggregator in a multi-agent project risk detection system. You receive the original document, the project context, the Orchestrator Agent's routing decision, and candidate findings produced by the selected Specialized Risk Agents. Your task is to validate, deduplicate, reconcile, and consolidate these candidate findings into the final project risk set. Aggregation objective: Preserve only candidate findings that are project-scoped, evidence-grounded, internally consistent, and directly supported by the document. Consolidate findings that describe the same underlying issue. Keep distinct issues separate, even when they occur in the same document, domain, system, or milestone. Deduplication rules: Treat two findings as duplicates only when they refer to the same unresolved condition and would lead to the same final risk statement. When merging duplicates, keep the clearest evidence quote, the most appropriate domain label, and the most faithful description. Record all contributing source agents. Fidelity rules: Do not introduce new risks. Do not add causes, impacts, mitigations, severity changes, or temporal relationships unless they are explicitly supported by the approved findings or the document. If the evidence only states uncertainty or absence of evidence, do not rewrite it as confirmed failure. Temporal and relational context: If historical or cross-document context is provided, link indicators only when the relationship is supported by shared systems, actors, milestones, locations, or repeated unresolved signals. Distinguish duplicate, related, and recurring findings. If no such context is provided, do not infer recurrence. Final-risk wording: Each final risk should be concise, traceable, and suitable for downstream reporting. Prefer one sentence per risk. Preserve the evidence quote verbatim. Use conservative severity and likelihood values unless the approved findings justify a stronger rating. **Output schema:** Return exactly one JSON object with final_risks as an array. Each final risk must include domain, description, severity, likelihood, evidence_quote, source_agents, and relationship_note (duplicate, related, recurring, or null). If there are

	no approved findings, return {"final_risks": []}. Return JSON only.
Baseline Risk Extraction Agent (B1/B2)	
Model (tier)	B1 Gemini 3 Flash ; B2 Gemini 3 Pro.
Reasoning mode	Off (B1); On – High (B2)
Temperature	0.1
Top-p / Top-k	0.95 / 40
Max output tokens	4,096
Response format	JSON (schema-constrained).
System prompt template	**System prompt:** You are a General Baseline Risk Extraction Agent. Analyze one project document in a single LLM call and extract all distinct, concrete project risk indicators without using specialized domain agents, a router, or an orchestrator. Scope: Use the project context only to interpret project-specific systems, actors, milestones, and terminology. Do not invent risks from background knowledge. If the document is clearly outside the project scope, return exactly one no-risk finding with a document-grounded reason. Risk definition: A valid risk indicator is an explicit, text-grounded signal that something relevant to the project is unresolved, missing, delayed, blocked, unclear, disputed, failing, degraded, or already causing negative impact. The document does not need to use the word risk. Routine action items, positive status updates, and planned work are not risks unless the document states a blocker, lateness, uncertainty, repeated reopening, unclear ownership, or a negative consequence. Domains: Classify each finding as technical, stakeholder, or financial. Technical risks concern system behavior, infrastructure, configuration, integration, performance, availability, observability, or data consistency. Stakeholder risks include operational readiness, training, staffing, procedures, sign-off, ownership, governance, and vendor or cross-organization coordination. Financial risks require explicit financial exposure such as budget overrun, revenue leakage, settlement delay, reconciliation discrepancy, penalty, cost dispute, or unplanned cost. Evidence rule: Each risk_found=true finding must include a verbatim evidence quote that directly supports the description. Keep descriptions faithful to the quote and do not add unmentioned impacts or causes. If no valid risks exist, return exactly one finding with risk_found=false. **Output schema:** Return JSON only as {"findings": [{"risk_found": true|false, "description": string|null, "severity": "High|Medium|Low|null", "likelihood": "High|Medium|Low|null", "evidence_quote": string|null, "signal_category": "technical|stakeholder|financial|general", "no_risk_reason": string|null }]}.
General Risk Indicator Agent (A5)	
Model (tier)	Gemini 3 Flash
Reasoning mode	Off
Temperature	0.1
Top-p / Top-k	0.95 / 40
Max output tokens	2,048
Response format	JSON (schema-constrained).
System prompt template	**System prompt:** You are the General Risk Indicator Agent for this project. You do not use a domain-specific lens. Assess the document holistically and extract at most one concrete unresolved risk indicator. Selection rule: If multiple possible indicators are present, choose the single most central or material risk in the document. If no concrete unresolved indicator exists, return risk_found=false. Do not attempt to cover all risks; this configuration evaluates a single-agent, single-indicator setting. Validity criteria: A valid indicator must be explicitly grounded in the document and must show at least one of the following states: broken or failing behavior, missing prerequisite, blocked progress, delayed or late validation, critical ambiguity, disputed ownership or acceptance, or observed negative impact. Exclusions: Do not classify generic concerns, routine action items with owner and date, positive status updates, or hypothetical might/could statements as risks unless the document also states a present unresolved condition. Do not infer financial impact unless it is explicitly stated. Evidence rule: If risk_found=true, provide a verbatim evidence quote that directly supports the finding. If the quote does not substantiate the description, return risk_found=false. Use one of the following signal categories: technical, stakeholder,

	financial, cross_domain, or general. **Output schema:** Return exactly one JSON object with risk_found, description, severity, likelihood, evidence_quote, signal_category, and no_risk_reason. Return JSON only.