MARUTI SUZUKI INDIA LIMITED

REQUEST FOR PROPOSAL

ANNEXURE A

MCP Server Development for MSIL

Phase: Development

ANNEXURE A

A. Key RFP Particulars:

S. No.	Particulars	Details
1.	RFP No.	RFQ97098
2.	Document Version	Version 1.0
3.	Date of issue of RFP	29th Jan 2026
4.	Last date for submission of queries	05th Feb 2026
5.	Pre-proposal submission meeting	TBD
6.	Last date for receipt of proposal	05th Feb 2026
7.	Date of Presentation from Parties (tentative)	TBD
9.	Contact Person (Name, address, email) (For submission of RFP Response)	Name: Kumar Vaibhav Singh Maruti Suzuki India Ltd. Email: kumar.singh@maruti.co.in Name: Antarish Deshpande Maruti Suzuki India Ltd. Email: antarish.deshpande@maruti.co.in

B. Glossary of Terms

Acronyms	Particulars
RFP	Request for Proposal
MSIL	Maruti Suzuki India Limited
IT	Information Technology
IPR	Intellectual Property Rights
MCP	Model Context Protocol
LLM/SLM	Large/Small Language Model
IAM / RBAC	Identity & Access Management and Role-Based Access Control
DLP	Data Loss Prevention controls
PIM	Privileged Identity Management
PAM	Privileged Access Management
OAuth 2.0	Open Authorization 2.0
OIDC	OpenID Connect
KMS	Key Management Service
HSM	Hardware Security Module
WAF	Web Application Firewall
IDS/IPS	Intrusion Detection System / Intrusion Prevention System
DLP	Data Loss Prevention
PII	Personally Identifiable Information
SAST/DAST	Static / Dynamic Application Security Testing

B. Purpose of this RFP

The purpose of floating this RFP is to evaluate different proposals with respect to MSIL's requirements. MSIL reserves the right, in its absolute discretion, to adopt any procurement strategy, following the evaluation of RFP responses, including (without limitation):

a. Direct negotiation with a single RFP respondent, or a shortlist of RFP respondents
b. Deciding not to proceed with any offer

C. Approach to Selection

Evaluation will be based on the relevance, clarity, and alignment of the proposed approach with DBP objectives. Based on the Technical Evaluation the vendor will be qualified for further rounds.

The responses provided will be used to determine:

a. Capacity and ability of respondent to deliver elements of the target product as per MSIL's requirements.
b. Ease of integration with MSIL network.
c. Support capability of the respondent.
d. Scalability and customization capability in the product.
e. Ease of onboarding of partners/users.
f. Security consideration.

D. Multistage Selection Process

MSIL will undertake a comprehensive, multi-stage evaluation to ensure technical robustness, commercial viability, and strategic alignment. MSIL reserves the right to modify timelines, seek clarifications, or discontinue the process at its discretion.

Refer Section 3.2 for details.

- **Stage 1 – RFP Issuance & Response Submission**

  - Issue RFP to invited partners.
  - Respondents submit **technical proposals**, including MCP Tool Definition Pack, Composite MCP Server architecture, and supporting documentation as explained in section 3.2.1

- **Stage 2 – Technical Evaluation & Demo**
  - **Demo to MSIL Team:** The demo must demonstrate that the MCP Client (agent/chatbot) can successfully call tools exposed by the Composite MCP Server using a generic live prompt related to service (e.g., booking a service appointment). It should handle dynamic inputs such as date and time and complete the transaction end-to-end.

    MSIL will validate the demo by checking database entries for successful transactions and performing code tracing in the MCP Server to confirm tool invocation logic and guardrails. Refer Section 3.2.2 for details.

- **Stage 3 – Management Pitch**
  - **Shortlisted vendors** present their solution approach, roadmap, and innovation highlights to MSIL leadership. Refer Section 3.2.3 for details.

- **Stage 4 – Commercial Submission**
  - Only vendors qualified in Stage 3 will be invited for commercial submission
  - Partners should propose Fixed-cost proposal aligned to Stage 1 deliverables; milestones & acceptance criteria; assumptions & exclusions; BOM (tools/licenses); SLA and support terms. Refer Section 3.2.4 for details.

- **Stage 5 – Commercial Negotiation & Award**
  - MSIL will negotiate with L1 (and/or others) prior to final award and onboarding.
  - Contract signing, onboarding plan, and governance kickoff. Refer Section 3.2.5 for details.

E. **Schedule of Events**

The following is a tentative schedule that will apply to this RFP, but may change in accordance with the MSIL's needs or unforeseen circumstances. Changes will be communicated by email to all invited respondents.

Stage	Step	Date, Time and Time Zone
Stage 1	Issue of RFP	29th Jan 2026
	Last date for RFP Response Submission	05th Feb 2026
Stage 2	Live Demo to MSIL Team	- Immediately after RFP Response Submission (TBD)
Stage 3	Pitch to Management	-TBD
Stage 4	Commercial Submission by Qualified Partners	-TBD
Stage 5	Commercial Negotiation with L1 and vendor onboarding	-TBD

Table of Contents

1 Executive Summary

Maruti Suzuki India Limited (MSIL) invites qualified partners to design, implement, and demonstrate a secure, efficient Composite MCP Server that enables conversational access to selected API Products (listed in the **Annexure**) from MSIL's API Manager—without any model training or RAG. Each selected API Product will be MCP-enabled through curated MCP tool definitions mapped to the product's OpenAPI/Swagger specification, allowing generic MCP clients (chatbots/agents) to discover and invoke business capabilities via structured tool calls.

MSIL's API Manager catalogs **more than 900 APIs** across business domains; **only the API Products explicitly listed in the Annexure are in scope for MCP enablement** in this phase. The architecture shall **keep a clean provision** to **add new Composite MCP Servers for future API Products** with minimal effort.

Example: "Enquiry" spans multiple specific journeys such as **test drive**, **new car**, **service**, **extended warranty**, and **MSIL Driving School (MDS)**; the Annexure will provide the API lists per journey.)

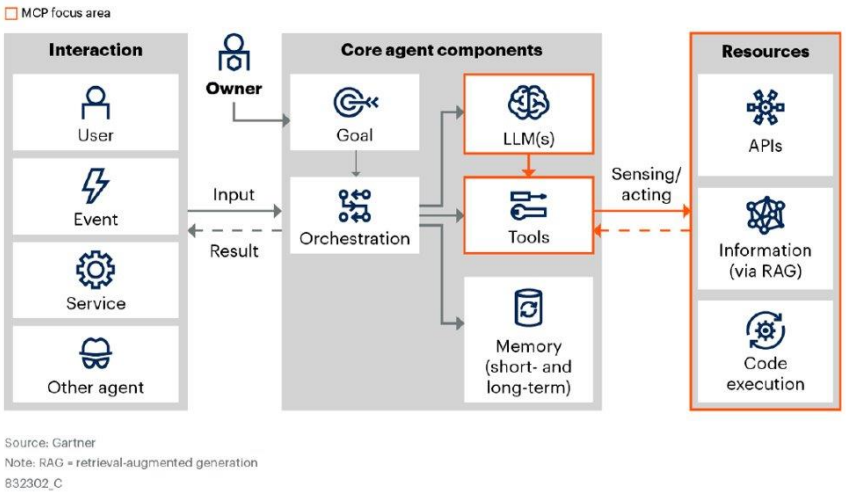

Fig 1.1: Logical Architecture and MCP focus Area

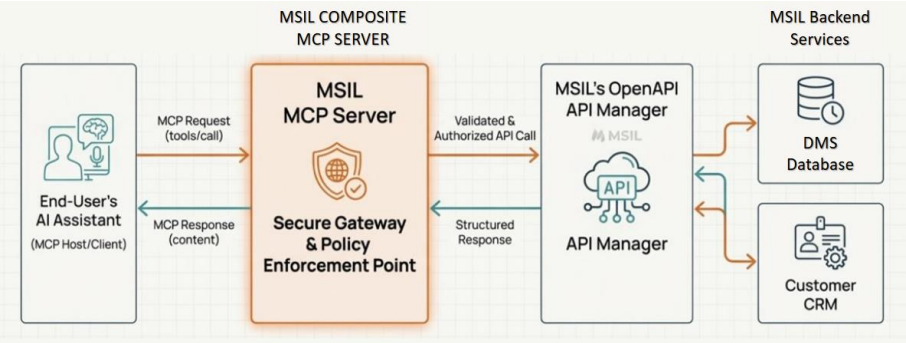

Fig 1.2: MCP Server Architecture

2 Background

This RFP is exclusively for **MCP server development**—including **MCP tool definition design**, server orchestration, security hardening, and integration with MSIL's **existing API Manager** (OpenAPI/Swagger) *refer fig 2.1*. The broad program intent is to **MCP-enable** our API landscape so **any compliant MCP client** (agent/chatbot) can safely discover and invoke business capabilities via standardized tools, with **low latency and minimal token overhead**.

Purpose:

- **MCP-enablement** of selected **API products** provided by MSIL (e.g., *Service Booking*, *Test Drive Booking*) to demonstrate **agentic tool use** via a **composite MCP server**.
- Establish an **MCP tool definition** library mapped to OpenAPI operations (verbs, paths, parameters, schemas) with **server-side validators, guardrails, and context controls** to mitigate misuse and **prompt injection**.
- Deliver a **repeatable reference** for scaling MCP enablement across additional API products with **DevSecOps** and observability baked in.

2.1 Current State (API Manager & MCP Readiness)

- **API Landscape:** MSIL operates a centralized **API Manager (Refer Fig 2.1)** with **Swagger/OpenAPI specifications** across multiple business domains. APIs already enforce **enterprise security policies (Refer Fig 2.2)** (e.g., authentication, authorization, network controls).
- **Consumption Gap:** Current agent/chatbot clients cannot **natively** consume these APIs via the **MCP standard**. There is **no unified tool-definition layer** or **composite MCP server** that abstracts business use cases as **discoverable, safe tools** with model-aware context limits.
- **Operational Considerations:** Existing observability, governance, and deployment processes are present on the API side but need **MCP-aware controls**, **rate-limit alignment**, **server-side validations**, and **cost-efficient context strategies** (e.g., caching, compact prompts, structured tool calls).

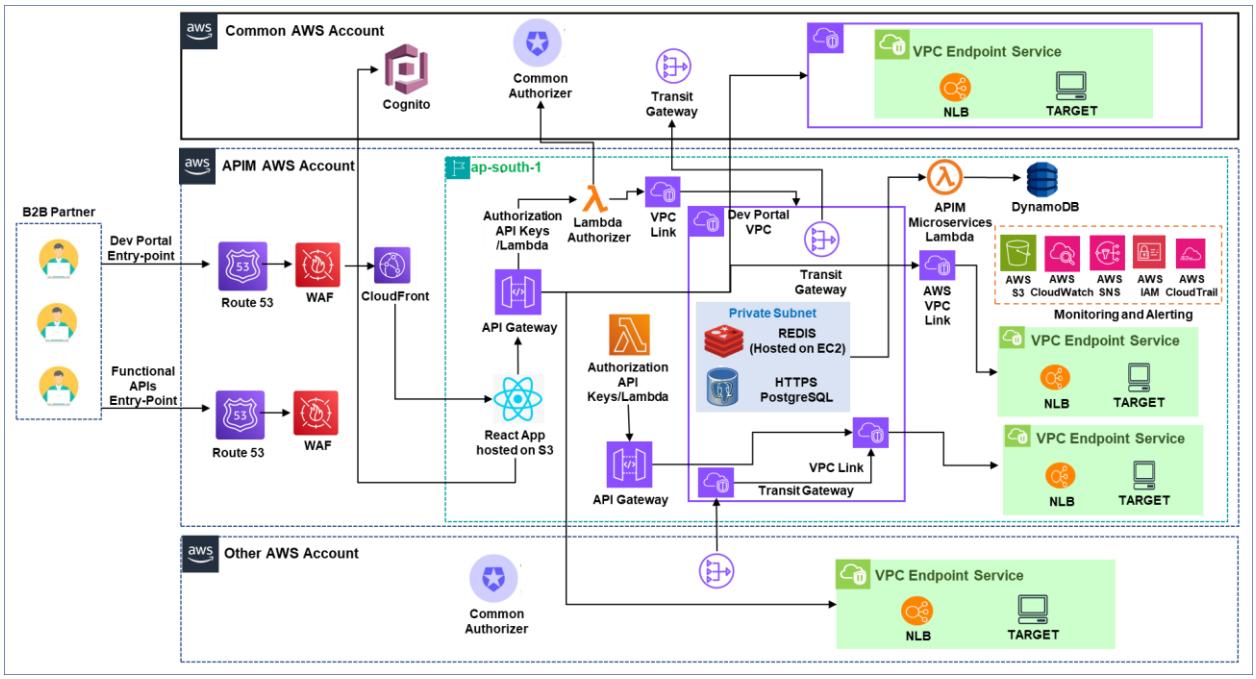

Fig 2.1: MSIL APIM Architecture

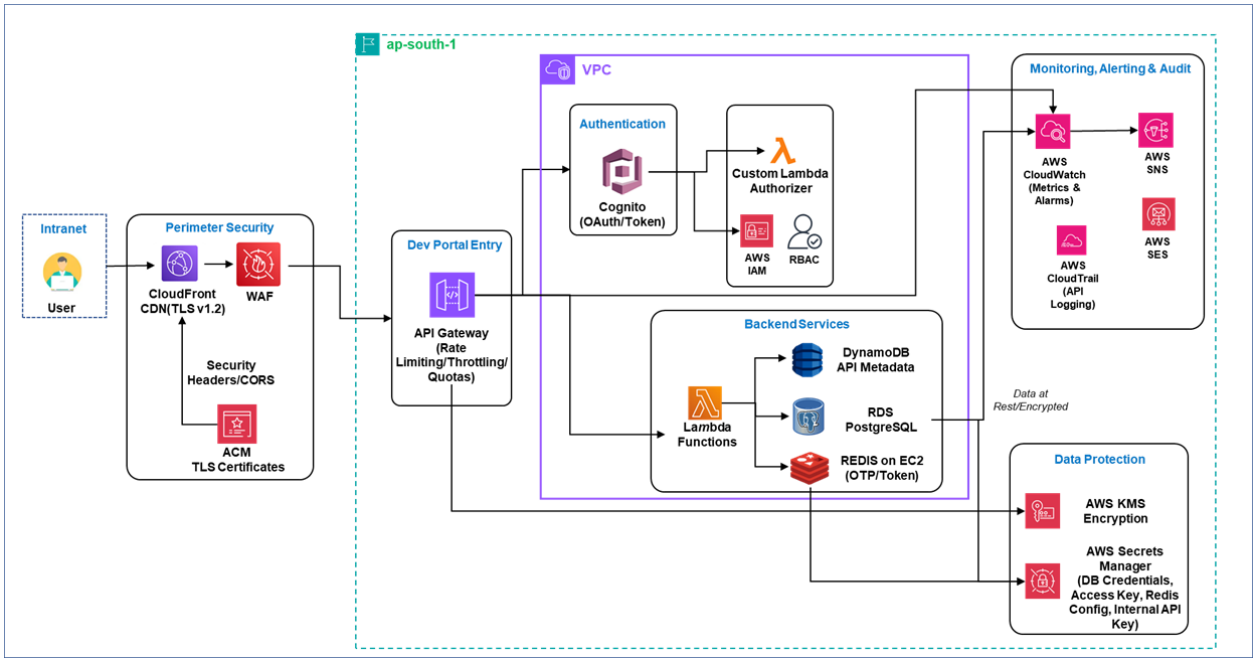

Fig 2.2: MSIL APIM Security Framework

2.2 Design Principles

1. **Composite MCP Architecture:** Prefer **Composite MCP Servers** tailored to concrete journeys to minimize tool sprawl and **reduce token burn**, while preserving central governance (registry/admin/policy) and AAA. (Refer Fig 2.3)
2. **Domain-Driven Topology & Central Discovery.** Treat each API Product as a **purposeful tool bundle** owned by a business/domain; register servers centrally for **governed discovery**, lifecycle control, and consistent semantics.
3. **Security-by-Design.** Enforce OAuth2/OIDC, RBAC (incl. **PIM/PAM**), strict schema validation, allowlists/denylists, secrets isolation, output sanitization, and rate-limit parity with API Manager.
4. **Observability & Audit.** Provide tamper-evident trails; standardized logs/metrics/traces for tool invocations and policy decisions; dashboards for reviewers.
5. **Efficiency & Cost Control.** Structured function calls, compact payloads, partial/streaming responses, and caching for idempotent reads to control token/latency overhead.
6. **Minimal Development Effort**

    - **OpenAPI-driven generation** of tool schemas and server stubs; **schema-first validation** (no bespoke parsing).
    - **Reusable templates/modules** for policies, guardrails, logging, and CI/CD.
    - **Standardized error models** and examples to accelerate testing and onboarding.
    - **Repeatable blueprint** to add future Composite MCP Servers for new API Products with minimal changes.
    - **Zero static coding**: the solution must bind dynamically to the **OpenAPI** definition, with **context and tool definition automatically updating**

2.3 Future State Architecture

Outcome Vision:

A **composite MCP server** that **adapts MSIL API products into MCP tools**, enabling **any compliant MCP client** to:

1. **Discover** tools (capabilities) per business use case;
2. **Validate & invoke** securely (policy-aligned);
3. **Operate efficiently** with **token-aware** server-side behaviours (structured arguments, compact responses, streaming where applicable, caching); and
4. **Observe & govern** (auditability, guardrails, rate limiting, incident response).

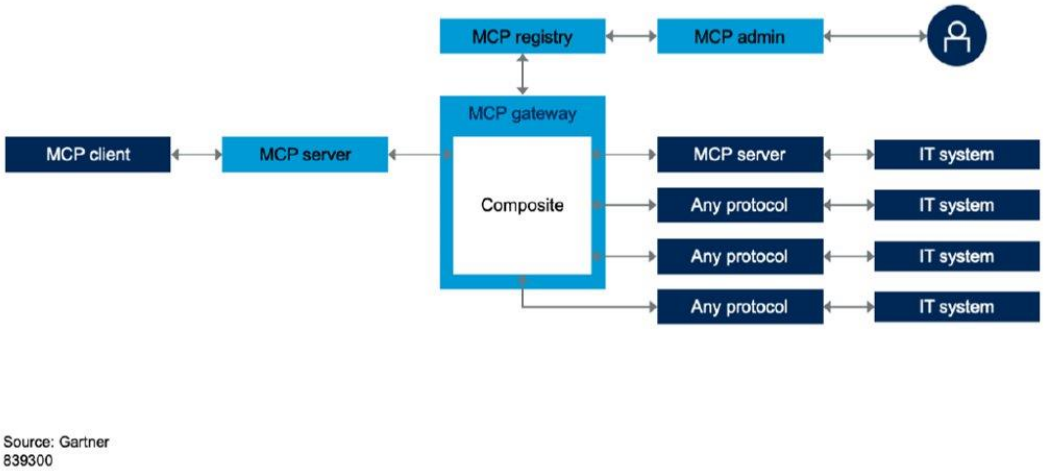

Fig 2.3: Composite MCP Architecture

Key Tenets:

- **Security-by-Design:** Strong IAM/RBAC alignment with API Manager; **prompt-injection** & **LLM/agent security** guardrails (input sanitization, allow-lists/deny-lists, schema validation, safe-response filters, secrets isolation).

- **Tool-First Architecture: MCP tool definitions** represent curated **API products in the annexure** (e.g., Service Booking) with **intent-aligned methods** (e.g., *CreateAppointment*, *GetSlots*, *UpdateBooking*), parameter schemas, and **server-side constraints** to **reduce token usage** (e.g., small inputs, partial responses, selective fields).

- **Composite Orchestration:** Multiple tool collections exposed via a single **composite MCP server** with **routing**, **policy enforcement**, and **shared guardrails** for consistent behavior.

- **DevSecOps & Observability:** Build pipelines, SAST/DAST, dependency scanning, IaC scans, structured logs/metrics/traces, audit trails of tool invocations and policy decisions.

- **Repeatable Enablement:** A **blueprint to onboard new API products** rapidly (design templates, tool schema checklist, testing harness, governance workflows).

3 Scope & Deliverables of RFP

3.1 Scope

- **MCP-enable API Products listed in the Annexure** via **Composite MCP Servers**. There are more than 900+ APIs in total out of which 309 APIs have to be converted into 30+ MCP Products (Final number to be decided by the partner during implementation).
- **Refer Composite MCP Server Architecture as shown in the Fig 2.3**
- **Tool-first conversational workflows** to be auto derived from **existing OpenAPI/Swagger definitions as the source of truth—no model training** and **no RAG**.
- **Minimal static coding**: the solution must bind dynamically to the **OpenAPI** definition, with **context and tool definition automatically updating**
- **Security-by-design & governance**: OAuth2/OIDC, RBAC/PIM/PAM parity with API Manager; central registration/discovery; full observability and audit.
- **Integration:** Seamless integration with MSIL's API Manager, CI/CD pipelines, and observability stack; deployment in containerized environments.
- **Scalability:** Architecture must allow rapid onboarding of new API Products using reusable templates and OpenAPI-driven generation.
- **Governance:** Provide dashboards for tool invocation metrics, error rates, and policy decisions; maintain audit logs for 12 months.

3.2 Deliverables

3.2.1 Stage 1: RFP Response Submission

All bidders must include the following components in their proposal covering all the points mentioned below:

1. **Proposed Solution Approach, Architecture and Demo Artefacts**
   - Recorded demo or secure link (if size exceeds submission limits).
   - Comprehensive **Architecture Documentation**, including:
     - **High-Level Architecture Diagram** – Logical view showing Composite MCP Server topology, integration with MSIL API Manager, and governance components.

- **Low-Level Technical Architecture** – Detailed component-level design (tool-definition layer, routing engine, validation modules, caching strategies, and observability stack).
- **Security Architecture** – IAM/RBAC/PIM/PAM enforcement, OAuth2/OIDC flows, schema validation, prompt injection guardrails, secrets isolation, and compliance posture (DPDP/GDPR).
- **Deployment Architecture** – Containerization (Docker/Kubernetes), CI/CD pipeline integration, environment segregation (Dev, QA, Prod), and DR/BCP strategy. (Leveraging MSIL APIM Architecture)
- **Integration Architecture** – API Manager connectivity, MCP tool registry, and central discovery mechanism.
- **MCP Tool Definition Artefacts: Auto-derived** from **OpenAPI/Swagger specifications** with validation rules, allowlists/denylists, error models, and positive/negative examples. (No Hard Coding/Static Coding)
- **Composite MCP Server Artefacts**: Curated tool catalog, policy and guardrails, identity integration, and full observability/audit capabilities.

2. **DevSecOps & Test Assets**
   - CI/CD pipeline artifacts (SAST/DAST, dependency/IaC scans), unit/integration tests, and negative test cases (unauthorized access, malformed requests, injection attempts).

3. **Observability Assets**
   - Structured logs, metrics, and traces; dashboards for tool invocations and policy decisions; tamper-evident audit trails.

4. **Documentation**
   - Architecture and design diagrams, API to Tool mapping tables (OpenAPI to MCP), deployment and operations runbooks.

5. **Team Structure**
   - Role-wise composition, named key personnel, relevant credentials, and escalation paths.

6. **Governance Templates**
   - Risk Register, Gap Register, User Stories, Project Status Reports, KPIs, and Program Management templates.

7. **Approach for MCP Product Selection**
   - Methodology for prioritizing and enabling API Products.(Domain/Journey)

8. **Implementation Roadmap & WBS**
    - Phases, milestones, dependencies, and timelines for Design, Discovery, and Development.

9. **Security Framework & Policy Management**
    - DevSecOps pipeline, release gates, compliance posture (DPDP/GDPR/CCI), and security controls.

10. **Support & SLA Management Process**
    - Support tiers, SLA definitions, incident workflows, escalation matrix, and monitoring/reporting cadence.

11. **Infrastructure, Software, Tools & Licences BOM to be provided along with the proposal** along with cost breakup.

3.2.2 Stage 2: Technical Evaluation & Demo

The bidder must provide a **live demo** demonstrating:

- An MCP Client (agent/chatbot) issuing a **generic prompt related to service booking** (e.g., “Book a service appointment for tomorrow at 10 AM”). *This prompt may vary but will be related to service booking only*
- The MCP Client must **discover and invoke tools via the Composite MCP Server** using structured tool calls as an output.
- The demo must handle **dynamic inputs such as date, time, mobile no, vehicle registration no., location/dealer** and complete the transaction end-to-end.
- **MSIL will validate the demo** by:
    - Checking **database entries** for successful transactions (e.g., service booking record created).
    - Performing **code tracing in the MCP Server** to confirm tool invocation logic and guardrails.
    - Reviewing **observability dashboards and audit logs** during the demo.
    - **Ensuring zero static coding**: the solution must bind dynamically to the **OpenAPI** definition, with **context and tool definition automatically updating** when the API specification changes.
- **Note:** MSIL will provide access to its **API Developer Environment** once the RFP is released.

3.2.3 Stage 3: Pitch Presentation

Qualified partners will be invited to present a **consolidated summary of outcomes from Stage 1 and Stage 2** to MSIL management.
The objective of this stage is to **validate executive confidence** in the partner's:

- **Solution Approach** – Architectural soundness, scalability, and alignment with MSIL's MCP enablement strategy.
- **Delivery Maturity** – Proven methodologies, resource readiness, and ability to meet timelines.
- **Governance Readiness** – Risk management, compliance posture, and program governance frameworks.
- **Execution Capability** – Demonstrated ability to deliver complex, secure, and efficient MCP server solutions at scale.

The presentation should be **concise, outcome-focused**, and include **key differentiators, innovation highlights, and risk mitigation strategies**.

3.2.4 Stage 4: Commercial Submission (Only if technically qualified)

Only vendors who qualify in Stage 2 (Technical Evaluation & Demo) and successfully complete Stage 3 (Management Pitch) will be invited to submit their **commercial proposal**.

Submission Requirements:

- **Fixed-Cost Proposal** aligned with Scope & Design Principles.
- **Detailed Cost Breakdown** including:
  - Bill of Materials (BOM) for tools, licenses, and infrastructure (if applicable).
  - Manpower cost by role and effort estimates.
  - Travel and incidental costs (if any).
- **Milestone-Based Payment Plan** linked to acceptance criteria.
- **Assumptions & Exclusions** clearly stated to define scope boundaries.
- **Support & SLA Terms** including response times, resolution timelines, and escalation matrix.

3.2.5 Stage 5: Commercial Negotiation & Award

Qualified partner (and/or other qualified partners at MSIL's discretion), shortlisted after Stage 4, will enter the **commercial negotiation phase** with MSIL.

Stage 5 will cover commercial closure, contractual finalization, and completion of vendor onboarding formalities. Final award is subject to successful negotiation closure, due diligence, and completion of MSIL's internal onboarding and compliance requirements.