MCP Server Compatibility and Implementation Guide for VS Code Agent Mode

This document provides a technical specification and developer guide for creating a Model Context Protocol (MCP) server that works with Visual Studio Code Agent Mode.

I. FORMAL MCP SERVER SPECIFICATION

---

1. Core Protocol

- Use JSON-RPC 2.0.
- Required message flow: initialize -> initialized.
- Include protocolVersion, capabilities, serverInfo in initialize response.
- Handle errors per JSON-RPC spec.

2. Supported Transports

- stdio: Required for local use.
- HTTP/SSE: Recommended for remote servers.
- WebSocket: Optional, for bi-directional messaging.

3. Initialization

VS Code sends initialize; server responds with protocolVersion and capabilities.

4. Mandatory Methods

- tools/list : List tools with metadata.
- tools/call : Execute named tool with args.
- notifications/tools/list_changed : Notify client when tools change.
- metadata/get : Return server name, version, capabilities.

5. Tool Definition

Each tool includes:

- name

- title

- description

- inputSchema (JSON schema defining parameters)

6. Workspace Roots

Server must respect workspace roots provided by VS Code.

7. Session Management

Maintain session context; handle multiple clients gracefully.

8. Security

- Tools disabled by default until user approval.

- Strict input validation.

- No arbitrary command execution.

9. Streaming

Optional but recommended (HTTP SSE or stdio).

II. IMPLEMENTATION GUIDE

---

1. Setup

- For local: communicate over stdin/stdout.

- For remote: use POST /mcp and GET /mcp (SSE).

2. Example Tool

Name: read_file

Description: Read a workspace file.

Input Schema: { "path": { "type": "string" } }

Output: File contents.

3. Tool Invocation

On tools/call, find handler and execute safely.

4. Metadata Endpoint

metadata/get should return { name, version, capabilities }.

5. Testing

- Run server.
- In VS Code: Command Palette -> Add MCP Server.
- Approve tools when prompted.
- Verify interaction.

III. CHECKLIST SUMMARY

[OK] JSON-RPC 2.0 compliant.

[OK] Supports stdio or HTTP/SSE.

[OK] Implements tools/list, tools/call.

[OK] Handles user consent for tools.

[OK] Secure and sandboxed.

[OK] Optional streaming supported.

[OK] Tested with VS Code Agent Mode.

Version: 2025.10

Sources: Microsoft DevBlogs, VS Code Docs, MCP Spec 2025-06-18.