



# Agentic SDLC Advanced

## Assignment:

Build a Minimal Model Context Protocol (MCP) Server for AI-Powered Data Injection.

## ⌚ Objective

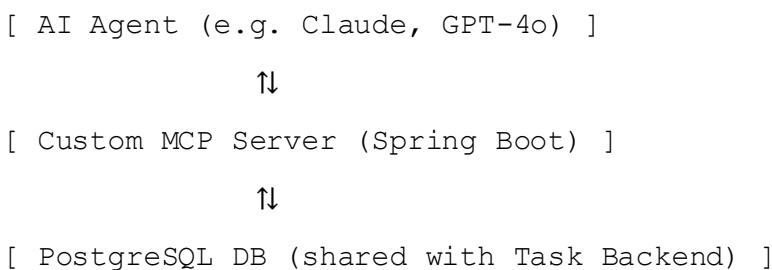
Develop a **MCP-compatible service in Spring Boot** that allows an **AI agent (e.g., Claude, GPT-4o, etc.)** to interact with the database of an existing Task Management application — specifically to:

- **Understand the schema**
- **Read from and insert into the database**
- **Automatically generate and insert realistic test data**

The MCP service acts as a **controlled access layer** between the agent and the database.

---

## ❖ Architecture



---

**Attention:** The MCP server is not part of the main application flow and is used only by the agent to read/write data.

---

## 📦 Project Scope

### Technologies

- **Spring Boot (Java 17+)**
  - **PostgreSQL**
  - **Spring Web + Spring Data JPA**
  - Optional: JSON Schema Generator or Swagger/OpenAPI
-



## Functional Requirements

### MCP Server Tools

Tool	Description
mcp-schema-tasks	Returns the database schema for the tasks table as a simplified JSON-Schema
mcp-tasks	Accepts a JSON array of Task objects and inserts them into the DB
mcp-tasks-summary	Returns summary statistics (e.g., task counts per status)
mcp-help	Returns a short, agent-readable description of available endpoints

---

## AI Agent Integration

The AI agent should be able to:

- Inspect the schema via *mcp-schema-tasks*
  - Generate 1000 realistic Task entries
  - Submit them to *mcp-tasks*
  - Validate success via *mcp-tasks-summary* or direct DB query
- 

## Success Criteria

1. **MCP Tool is running and accessible and respects specification version 2025-06-18<sup>i</sup>**
  2. **Schema inspection works via *mcp-schema-tasks***
  3. **AI agent inserts 1000 task records via *mcp-tasks***
  4. **Summary endpoint reflects the inserted data (e.g., shows 1000 tasks split across statuses)**
  5. **All actions are documented**, including prompts, AI output, and results
-



## 📋 Deliverables

- **Git Repository** with:
    - MCP Spring Boot source code
    - DB access setup (e.g., shared PostgreSQL connection config)
  - **Documentation** (README.md) including:
    - How the MCP server works
    - Sample prompts used with the agent
    - Result of the test: inserting 1000 records
    - Screenshot or log output showing inserted data count
- 

## 💬 Prompt Example for the Agent

"Please inspect the task schema at /mcp/schema/tasks. Then generate and insert 1000 diverse tasks with random statuses, titles, and due dates using the /mcp/tasks endpoint."

---

## 📝 Optional Enhancements

- OpenAPI documentation (/swagger-ui)
- Auth token for AI interaction (if needed)
- Logging or auditing of AI-originated actions

---

<sup>1</sup> <https://modelcontextprotocol.io/specification/2025-06-18>