

**Project Title:** Development and Integration of MCP Servers for Advanced Storage Systems

**Objective:** To enhance the research assistant's (RA) proficiency in Model Context Protocol (MCP) server development, enabling the creation and integration of new MCP servers tailored for emerging storage systems.

**Project Description:** This project is structured to provide the RA with comprehensive exposure to MCP server development, from foundational understanding to practical implementation and testing. The RA will engage in a series of tasks designed to build technical competence and ensure the seamless integration of MCP servers with storage solutions. A strong emphasis is placed on meticulous documentation to facilitate knowledge sharing and collaborative development.

### **Tasks and Timeline:**

#### **Week 1: Foundation and Exploration**

##### **1. Study Model Context Protocol (MCP):**

- **Objective:** Acquire a solid understanding of MCP, its architecture, and its applications.
- **Activities:**
  - Review the MCP Core Architecture to grasp its foundational concepts.
  - Explore the MCP Client Development Guide for insights into client-server interactions.
  - Consult the Building MCP Servers: Part 1 article for practical insights into server development.

##### **2. Examine Existing MCP Tasks:**

- **Objective:** Analyze current MCP tasks to understand practical implementations and identify areas for improvement.
- **Activities:**
  - Review existing MCP server implementations, focusing on their functionalities and integration strategies.
  - Identify potential enhancements or adaptations applicable to new storage systems.

#### **Week 2: Implementation and Testing**

### 3. Deploy Filesystem and Auxiliary MCP Servers:

- **Objective:** Set up and configure MCP servers pertinent to filesystem operations and additional functionalities.
- **Activities:**
  - Utilize the MCP Server Step-by-Step Guide to facilitate server setup.
  - Ensure compatibility with existing storage infrastructures.

### 4. Test Minimum Viable Product (MVP) Server with LLM Calls:

- **Objective:** Validate the operational readiness of the MVP server using a comprehensive suite of Large Language Model (LLM) calls.
- **Activities:**
  - Develop and execute a series of LLM calls that encompass all server functionalities.
  - Document test outcomes, noting any discrepancies or areas requiring optimization.

## Week 3: Integration and Optimization

### 5. Explore Python Libraries for MCP Development:

- **Objective:** Investigate Python libraries that facilitate MCP server creation and management.
- **Activities:**
  - Examine the MCP Python SDK for relevant tools and frameworks.
  - Assess how these libraries can streamline server development processes.

### 6. Adapt Servers to Utilize Python Libraries:

- **Objective:** Refactor existing servers, initially built on JSPN, to integrate Python-based MCP libraries.
- **Activities:**
  - Modify server architectures to incorporate Python libraries, enhancing functionality and maintainability.

- Ensure that adaptations align with project objectives and performance benchmarks.

## 7. Retest Adapted Servers:

- **Objective:** Confirm the operational integrity of the adapted servers through rigorous testing.
- **Activities:**
  - Re-execute the LLM test suite developed in Task 4.
  - Compare results to previous benchmarks, ensuring all functionalities perform as intended.

## Documentation and Knowledge Sharing:

Throughout the project, the RA is expected to:

- **Maintain Comprehensive Documentation:**
  - **Objective:** Create detailed records of methodologies, configurations, code annotations, and testing procedures.
  - **Activities:**
    - Develop a structured README.md file outlining project objectives, setup instructions, and usage guidelines.
    - Utilize GitHub's Wiki feature to host extended documentation, including design decisions and troubleshooting guides.
- **Utilize GitHub for Continuous Integration:**
  - **Objective:** Leverage GitHub's features for version control, collaboration, and project management.
  - **Activities:**
    - Implement a clear branching strategy to manage development workflows.
    - Use GitHub Projects to track progress, set milestones, and manage tasks.
    - Ensure all code and documentation are regularly committed and pushed to the repository.

**Expected Outcomes:**

- Enhanced proficiency in MCP server development and integration.
- Operational MCP servers tailored for new storage systems.
- A well-documented project repository facilitating future development and knowledge transfer.

**Support and Resources:**

- **Mentorship:** Regular consultations with senior developers or mentors to provide guidance and address challenges.
- **Learning Materials:** Access to relevant tutorials, documentation, and community forums to support learning and problem-solving.

By adhering to this structured plan, the RA will develop the necessary skills and knowledge to contribute effectively to MCP server development projects, ensuring both personal growth and the advancement of organizational objectives.