

Universidad del Valle de Guatemala

Facultad de Ingeniería Ciencias de la Computación



MCP Server

Redes

Angel Herrarte

28 de agosto del 2025

Se propone el desarrollo de un servidor MCP (Model Context Protocol) especializado en el análisis profundo de repositorios Git, que proporcione métricas avanzadas de calidad de código, detección de patrones problemáticos, análisis de contribuciones y generación de reportes comprehensivos. Este servidor permitirá a los desarrolladores obtener insights valiosos sobre la salud técnica de sus proyectos mediante el procesamiento inteligente de la historia del repositorio, análisis estático del código y aplicación de algoritmos de detección de anomalías.

A diferencia de funcionalidades triviales como consultar el clima, este servidor MCP implementa:

- **Análisis multi-dimensional:** Examina código, historia, contribuciones y patrones simultáneamente
- **Algoritmos complejos:** Implementa detección de code smells, cálculo de complejidad ciclomática, y análisis de acoplamiento
- **Procesamiento de datos masivos:** Capaz de analizar repositorios con miles de commits y archivos
- **Inteligencia contextual:** Genera recomendaciones personalizadas basadas en el contexto del proyecto
- **Integración con herramientas externas:** Conecta con servicios de análisis estático y linters especializados

Análisis de Complejidad

Cálculo de complejidad ciclomática, cognitiva y análisis de mantenibilidad del código

Detección de Code Smells

Identificación automática de antipatrones y problemas de diseño en el código

Métricas de Evolución

Análisis temporal de cómo ha evolucionado la calidad del código a lo largo del tiempo

Análisis de Contribuciones

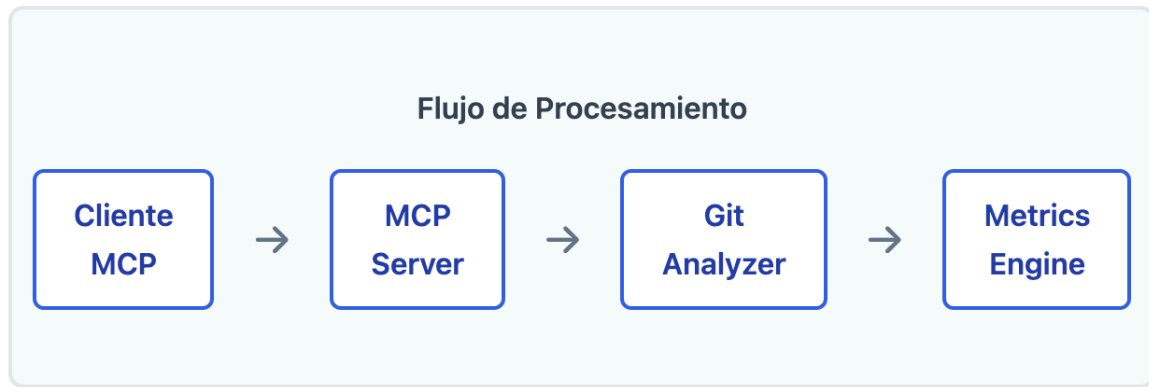
Estadísticas detalladas sobre contribuyentes, frecuencia de commits y ownership del código

Análisis Arquitectural

Detección de dependencias circulares, análisis de acoplamiento y cohesión

Hotspot Detection

Identificación de archivos problemáticos que requieren refactorización urgente



Componentes Principales

- **Git Interface:** Módulo de interacción con repositorios locales y remotos
- **Parser Engine:** Análisis sintáctico para múltiples lenguajes (Python, JavaScript, Java, etc.)
- **Metrics Calculator:** Motor de cálculo de métricas de calidad
- **Pattern Detector:** Sistema de detección de patrones y antipatrones
- **Report Generator:** Generador de reportes en formatos JSON, HTML y Markdown

Método	Descripción	Parámetros
analyze_repository	Análisis completo del repositorio	repo_path, branch, depth
get_code_metrics	Obtiene métricas de calidad del código	file_path, metric_types[]
detect_smells	Detecta code smells y antipatrones	repo_path, sensitivity_level
analyze_contributors	Análisis de contribuciones y ownership	repo_path, time_range
get_hotspots	Identifica archivos problemáticos	repo_path, threshold
generate_report	Genera reporte comprehensivo	analysis_id, format, sections[]

Tecnologías a Utilizar

- **Lenguaje:** Python 3.10+
- **Framework MCP:** Implementación del protocolo MCP estándar
- **Librerías principales:**
 - GitPython - Interacción con repositorios Git

- AST (Abstract Syntax Tree) - Análisis de código Python
- Radon - Cálculo de métricas de complejidad
- Pygments - Análisis léxico multi-lenguaje
- NetworkX - Análisis de grafos de dependencias