MCP UNIVERSE - Universal Playground for Model Context Protocols

Project Overview

MCP UNIVERSE is a revolutionary platform designed to make traditional APIs obsolete by enabling AIs to dynamically discover, negotiate, and compose solutions from the best available tools across the entire universe in real-time. It provides a universal playground for Model Context Protocols (MCP) that enables dynamic tool discovery, execution, and sharing.

Key Features

- **Universal MCP Support**: Connect to any MCP server via SSE, STDIO, Docker, or HTTP.
- **Intelligent Routing**: AI-driven orchestration with trust-based scoring and adaptive routing.
- **Trust-based Selection**: Decentralized reputation management for providers.
- **Session Sharing**: Immutable session storage with shareable URLs and embeddable widgets.
- **Real-time Execution**: Concurrent tool execution support with real-time performance monitoring.

Architecture

The MCP UNIVERSE is built with a modular, microservices-oriented architecture, consisting of the following core components:

1. **Nexus Gateway (Rust)**: Primary entry point for all connections, handling various transport protocols, authentication, and rate limiting.
2. **Intelligent Router & Orchestrator (Rust)**: Dynamic tool discovery and selection, utilizing a trust-based scoring algorithm and adaptive routing logic.
3. **Trust Graph Engine (PostgreSQL)**: Decentralized reputation management system, storing provider scores, attestations, and implementing fraud detection.
4. **Universal Client Library (TypeScript)**: A browser-based MCP client for connecting to the gateway, rendering tool calls, and managing sessions.
5. **Session Management System (Flask)**: Handles storage and retrieval of shared sessions, using Redis for ephemeral data and S3-compatible storage for artifacts.
6. **Frontend (React)**: A minimalist, responsive UI built with Next.js 14 and Tailwind CSS, providing a chat-style interface for interaction.

Project Structure

```
/home/ubuntu/mcp-universe/
├── MANIFESTO.md                    # Project Vision
├── PHILOSOPHY.md                   # Philosophy and Principles
├── nexus-core/                     # Rust Gateway implementation
├── intelligent-router/             # Rust Router & Orchestrator implementation
├── trust-graph-engine/             # PostgreSQL Trust Graph Engine
implementation
├── universal-client-library/       # TypeScript Universal Client Library
├── session-management-system/      # Flask Session Management System
├── mcp-universe-frontend/          # React Frontend application
└── docker-compose.yml              # Docker Compose configuration for the
entire system
```

Setup and Running with Docker Compose

To set up and run the entire MCP UNIVERSE system using Docker Compose, follow these steps:

1. **Navigate to the project root:** `bash cd /home/ubuntu/mcp-universe`

2. **Build and start the services:** `bash docker-compose up --build -d` This command will:

    - Build the Docker images for `nexus-gateway`, `session-manager`, and `frontend`.
    - Start all services (PostgreSQL, Redis, Nexus Gateway, Session Manager, Frontend) in detached mode.

3. **Access the Frontend:** Once all services are up and running, you can access the MCP UNIVERSE frontend in your web browser at `http://localhost:80`.

4. **Access Backend Services (for testing/development):**

    - Nexus Gateway: `http://localhost:3000`
    - Session Management System: `http://localhost:5001`
    - PostgreSQL: `localhost:5432`
    - Redis: `localhost:6379`

Development

Each component has its own `README.md` and development instructions within its respective directory. Refer to those for detailed development guides.

License