

API & Services Overview

Purpose: Centralized overview of all APIs and services for the moOde Audio project

Last Updated: 2025-01-03

■ Current Status

■ Active Services (No Setup Required)

1. **Filesystem MCP** - File operations
2. **SQLite MCP** - moOde database access
3. **Puppeteer MCP** - Browser automation (testing Web UI)
4. **Fetch MCP** - HTTP requests
5. **Memory MCP** - Persistent memory/context

■ Available but Need API Keys

6. **GitHub MCP** - Access GitHub repositories
 - **Status:** Configured, needs token
 - **Setup:** `GITHUB_PERSONAL_ACCESS_TOKEN` in `.env`
 - **Useful for:** Reading moOde source, checking updates, issues/PRs
7. **Brave Search MCP** - Web search
 - **Status:** Configured, needs key
 - **Setup:** `BRAVE_API_KEY` in `.env`
 - **Useful for:** Searching moOde forum, finding solutions, research
8. **PostgreSQL MCP** - PostgreSQL database
 - **Status:** Configured, needs connection string
 - **Setup:** `POSTGRES_CONNECTION_STRING` in `.env` (if needed)
 - **Useful for:** Database operations (moOde uses SQLite, so may not be needed)

■ Recommended Services to Consider

For Development Work (Help Me Help You)

1. **GitHub MCP** ■ **HIGH PRIORITY**
 - Helps me read moOde source code directly
 - Check for updates and fixes
 - Access issues and pull requests
 - **Free:** GitHub Personal Access Token (no cost)
2. **Brave Search MCP** ■ **MEDIUM PRIORITY**
 - Search moOde forum and documentation
 - Find solutions to problems
 - Research hardware/software issues
 - **Cost:** Check Brave Search API pricing

For moOde Audio Project (Future Integrations)

3. ****Music Metadata APIs**** (optional)

- Last.fm API (already used in some drivers)
- MusicBrainz API
- Discogs API
- ****Use case:**** Enhanced metadata, artwork, music information

4. ****Cloud Storage APIs**** (optional)

- Google Drive API
- Dropbox API
- OneDrive API
- ****Use case:**** Stream music from cloud storage

5. ****Streaming Services**** (optional)

- Spotify API
- Apple Music API
- ****Use case:**** Integration with streaming services

■ Setup Instructions

Step 1: Create `.env` File Template

Create `.env` in project root (if not exists):

```
# GitHub MCP
GITHUB_PERSONAL_ACCESS_TOKEN=your_token_here

# Brave Search MCP
BRAVE_API_KEY=your_key_here

# PostgreSQL MCP (if needed)
POSTGRES_CONNECTION_STRING=postgresql://user:pass@host/db
```

Step 2: Get API Keys

GitHub Personal Access Token (Free)

1. Go to: <https://github.com/settings/tokens>
2. Click "Generate new token" → "Generate new token (classic)"
3. Select scopes:
 - ``repo`` (full repository access)
 - ``read:org`` (if accessing org repos)
4. Copy token and add to `.env``

Brave Search API Key

1. Go to: <https://brave.com/search/api/>
2. Sign up/login
3. Get API key
4. Add to `.env``

■ How to Work Together

For Adding New APIs/Services:

1. ****Tell me what you want:****
 - "I want to add [Service Name] API"
 - "I need access to [Service] for [purpose]"
2. ****I will:****
 - Check if it's available as MCP server
 - Show you how to get API key
 - Help configure it
 - Update this documentation
3. ****You provide:****
 - API key/token
 - Clear purpose/goal
 - Any specific requirements

Template for Requesting Services:

```
Service: [Name]
Purpose: [What you want to do with it]
Priority: [High/Medium/Low]
Cost: [Free/Paid/Unknown]
```

■ Quick Checklist

- ☐ GitHub MCP configured? (Recommended)
- ☐ Brave Search MCP configured? (Optional but useful)
- ☐ `.env` file exists and is properly formatted?
- ☐ API keys added to `.env`?
- ☐ Tested that services work?

■ Related Documentation

- `docs/MCP_SERVERS_CONFIGURED.md` - Detailed MCP server info
- `docs/MCP_SERVERS_SETUP.md` - Setup instructions
- `.cursorrules` - Project rules and structure

■ Next Steps

1. ****Decide which services you want:****
 - GitHub MCP: Recommended (free, very useful)
 - Brave Search: Optional (helps with research)
2. ****Get API keys:****
 - Follow setup instructions above
3. ****Tell me:****
 - "Set up GitHub MCP with my token: xxxxx"
 - "I want to add [service]"

4. ****I'll help:****

- Configure everything
- Test it works
- Update documentation

Remember: You're practicing working with me! Give me clear instructions, and I'll help you set everything up perfectly. ■