

Troubleshooting Guide

This guide covers common issues and their solutions for AI Orchestrator.

Table of Contents

- [MCP Server Issues](#)
 - [Xcode Extension Issues](#)
 - [API and Model Issues](#)
 - [Installation Issues](#)
-

MCP Server Issues

Server Logs Are Empty

Symptoms: Starting the server shows success but `logs/mcp-server.log` is empty or doesn't exist.

Cause: MCP servers communicate via stdio (stdin/stdout), not traditional logging. The previous version didn't have file-based logging.

Solution:

1. Update to the latest version which includes file-based logging
2. Logs are now written to `logs/mcp-server.log`
3. Check the log file:

```
bash
tail -f logs/mcp-server.log
```

Server Starts But Cursor Can't Connect

Symptoms: The server process is running but Cursor IDE doesn't show the tools.

Cause: MCP servers need proper configuration in Cursor settings.

Solution:

1. Check your Cursor MCP configuration file:

- Linux/Mac: `~/.cursor/mcp.json`
- Or via Cursor Settings → MCP

1. Add the server configuration:

```
json
{
  "servers": {
    "ai-orchestrator": {
      "command": "/path/to/ai_orchestrator/venv/bin/python",
      "args": ["-m", "mcp_server.server"],
      "cwd": "/path/to/ai_orchestrator",
      "env": {
        "PYTHONPATH": "/path/to/ai_orchestrator"
      }
    }
  }
}
```

```

    }
  }
}

```

2. Restart Cursor IDE

“Module not found” Errors on Server Start

Symptoms: Server fails with `ModuleNotFoundError: No module named 'mcp'` or similar.

Solution:

1. Make sure dependencies are installed:

```

bash
cd /path/to/ai_orchestrator
source venv/bin/activate
pip install -r requirements.txt
pip install mcp

```

1. Verify installation:

```

bash
python -c "import mcp; import ai_orchestrator; print('OK')"

```

Server Crashes on Startup

Symptoms: Server exits immediately with an error.

Solution:

1. Run the server in foreground mode to see errors:

```

bash
./scripts/start-server.sh --fg

```

1. Check the log file for detailed errors:

```

bash
cat logs/mcp-server.log

```

2. Common issues:

- Missing `.env` file: Copy `.env.example` to `.env`
- Invalid API keys: Verify your API keys in `.env`
- Python version: Requires Python 3.10+

Xcode Extension Issues

“No such module ‘XcodeKit’” Build Error

Symptoms: Building with `swift build` fails with:

```
error: no such module 'XcodeKit'
```

Cause: XcodeKit is only available when building as an Xcode Source Editor Extension within a proper Xcode project. Swift Package Manager cannot access this framework.

Solution:

1. Use the provided build script that generates a proper Xcode project:

```
bash
cd xcde_extension
./Scripts/build.sh --generate
```

1. **Never** use `swift build` directly for the extension.
2. If you need to build manually:

```
bash
./Scripts/generate-xcode-project.sh
open AIOrchestratorXcode.xcodeproj
# Build from Xcode
```

Extension Doesn't Appear in Xcode

Symptoms: After building, the extension doesn't show up in Xcode's Editor menu.

Solution:

1. Enable the extension in System Settings:
 - Open **System Settings** → **Privacy & Security** → **Extensions**
 - Find "Xcode Source Editor" category
 - Enable "AI Orchestrator Extension"
1. Restart Xcode completely (Cmd+Q, then reopen)
2. If still not working, try:

```
bash
# Reset extension cache
killall Xcode
rm -rf ~/Library/Caches/com.apple.dt.Xcode
```

Extension Commands Are Grayed Out

Symptoms: The extension appears in the menu but commands are disabled.

Cause: Some commands require a source file to be open or text to be selected.

Solution:

1. Make sure you have a Swift/Objective-C file open
2. For commands like "Explain Code", select some text first
3. Check Console.app for extension errors:

```
bash
# Open Console.app and filter by "AI Orchestrator"
```

Build Succeeds But Extension Doesn't Load

Symptoms: Xcode builds successfully but the extension fails to load.

Solution:

1. Check for code signing issues:

```
bash
# Build without code signing for development
./Scripts/build.sh --clean
```

1. Check Console.app for loading errors

2. Try running from Xcode directly:

- Open the project in Xcode
- Select the extension scheme
- Run (⌘R) to debug

API and Model Issues

“No models configured” Error

Symptoms: Orchestrator returns error about no available models.

Solution:

1. Check your `.env` file has valid API keys:

```
bash
```

```
cat .env | grep -E "(OPENAI|ANTHROPIC|GEMINI|MOONSHOT)_API_KEY"
```

1. At least one API key must be configured:

```
env
```

```
OPENAI_API_KEY=sk-...
```

```
ANTHROPIC_API_KEY=sk-ant-...
```

```
GEMINI_API_KEY=...
```

2. Test API connectivity:

```
bash
```

```
./scripts/test-apis.sh
```

API Rate Limit Errors

Symptoms: Requests fail with rate limit or quota errors.

Solution:

1. Check your API plan limits
2. Add delay between requests
3. Use a different model provider as fallback

Model Response Errors

Symptoms: AI responses are cut off or return errors.

Solution:

1. Check token limits in your configuration
2. Reduce input size for large codebases
3. Try a different model:

```
env
```

```
ANTHROPIC_MODEL=claude-3-5-sonnet-20241022
```

```
OPENAI_MODEL=gpt-4o-mini
```

Installation Issues

Virtual Environment Not Found

Symptoms: Scripts fail with “venv not found” error.

Solution:

```
cd /path/to/ai_orchestrator
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

Permission Denied Errors

Symptoms: Scripts fail with permission errors.

Solution:

```
# Make scripts executable
chmod +x scripts/*.sh
chmod +x install.sh uninstall.sh update.sh
```

Missing Dependencies on macOS

Symptoms: Build fails with missing system dependencies.

Solution:

```
# Install Xcode command line tools
xcode-select --install

# Update Homebrew packages if needed
brew update
brew install python@3.11
```

Getting More Help

If you're still experiencing issues:

1. **Check logs:**

```
bash
tail -100 logs/mcp-server.log
```

2. **Enable debug mode:**

```
bash
export DEBUG=1
./scripts/start-server.sh --fg
```

3. **Open an issue** on GitHub with:

- Your OS and version
- Python version (`python --version`)
- Xcode version (if applicable)

- Relevant log output
- Steps to reproduce

Common Error Messages Reference

Error	Likely Cause	Solution
<code>ModuleNotFoundError: No module named 'mcp'</code>	Missing dependency	<code>pip install mcp</code>
<code>no such module 'XcodeKit'</code>	Using SPM instead of Xcode	Use <code>./Scripts/build.sh --generate</code>
<code>OPENAI_API_KEY not set</code>	Missing <code>.env</code>	Copy <code>.env.example</code> to <code>.env</code>
<code>Connection refused</code>	Server not running	Run <code>./scripts/start-server.sh</code>
<code>Invalid API key</code>	Wrong API key format	Check API key in <code>.env</code>