

Nimble

AI-powered Adaptive Project and Work Management



Connecting the NimbleWork MCP to Codex (CLI)

Date: 21-OCT-2026

Copyright Information

Copyright © 2026 NimbleWork, Inc. Any use of this material, including reproduction in whole or in part, requires permission in writing from NimbleWork, Inc. Therefore, and unless otherwise specified, no part of this document may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, microfilm, or scanning, without permission in writing from NimbleWork, Inc.

Table of Contents

[Connecting the NimbleWork MCP to Codex \(CLI\)](#)

[1. Purpose](#)

[2. Prerequisites](#)

[3. Connect the NimbleWork MCP to Codex \(CLI\)](#)

[Step 1 — Sign in to Codex](#)

[Step 2 — Add the NimbleWork MCP server](#)

[Step 3 — Authorize the connector](#)

[Step 4 — Use it](#)

[Step 5 — Troubleshooting](#)

[4. Support](#)

[NimbleWork MCP Connector](#)

Connecting the NimbleWork MCP to Codex (CLI)

Product: Nimble (enterprise agile planning & flow-management platform)

Connector: NimbleWork MCP

1. Purpose

This guide walks you through connecting the NimbleWork MCP connector to Codex — OpenAI's command-line tool — so you can work with your Nimble projects through natural language, directly from the terminal. The steps below use Windows Command Prompt (cmd).

2. Prerequisites

- The Codex CLI already installed on your system (run `codex --version` to confirm).
- An OpenAI / ChatGPT account that can run Codex.
- Your Nimble account credentials.
- Access to the Nimble workspace(s) you want to work with.
- Basic familiarity with running commands in Command Prompt (cmd).

3. Connect the NimbleWork MCP to Codex (CLI)

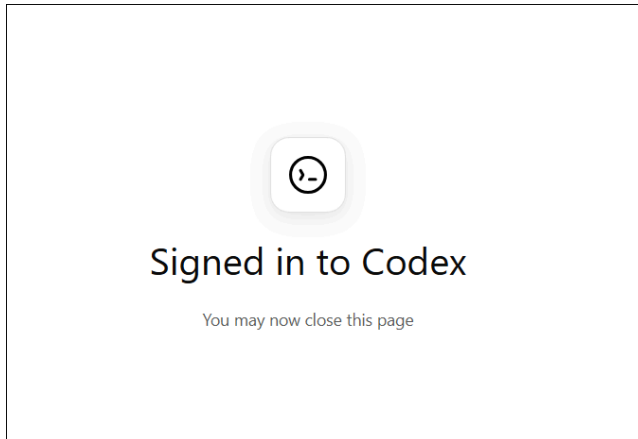
You sign in to Codex, add the NimbleWork server to your Codex configuration, then authorize it in your browser. All commands below are run in Command Prompt (cmd).

Step 1 — Sign in to Codex

If you are not already signed in, run:

```
codex login
```

This opens your browser — sign in with your ChatGPT (OpenAI) account, then return to the terminal.



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.26200.9711]
(c) Microsoft Corporation. All rights reserved.

C:\Users\>codex login
Starting local login server on http://localhost:1455.
If your browser did not open, navigate to this URL to authenticate:

https://auth.openai.com/oauth/authorize?response_type=code&client_id=app_EMoamEEZ73f0CkXaXp7hrann&redirect_uri=http%3A%2F%2F127.0.0.1%3A1455%2Fcallback&code_challenge=aSc6um9YfwtKKqBF4BhvRnjDK8y4GwmFlTX119qjw0c&code_challenge_method=S256&state=fTFxj-Z4LtrusqBF8cYQaU-FZ84tND73uMyeFgBNAhs&scope=openid+profile+email+offline_access+api.connectors.read+api.connectors.invoke&id_token_add_organizations=true&codex_cli_simplified_flow=true&originator=codex_cli_rs

On a remote or headless machine? Use `codex login --device-auth` instead.
Successfully logged in

C:\Users\>

```

Step 2 — Add the NimbleWork MCP server

Add the NimbleWork server to your Codex configuration file. Open (or create) the file at:

```
%USERPROFILE%\codex\config.toml
```

and add the following entry:

```
[mcp_servers.nimblework-mcp]
```

```
url = "https://api.digite.com/mcp/v1"
```

```
[mcp_servers.nimblework-mcp.oauth]
```

```
client_id = "chatgpt_prod"
```

```
callback_url = "http://localhost:5555/callback/JCtaWCXyXR5e"
```

```
callback_port = 5555
```

```
1 [mcp_servers.nimblework-mcp]
2 url = "https://api.digite.com/mcp/v1"
3
4 [mcp_servers.nimblework-mcp.oauth]
5 client_id = "chatgpt_prod"
6 callback_url = "http://localhost:5555/callback/JCtaWCXyXR5e"
7 callback_port = 5555
8
```

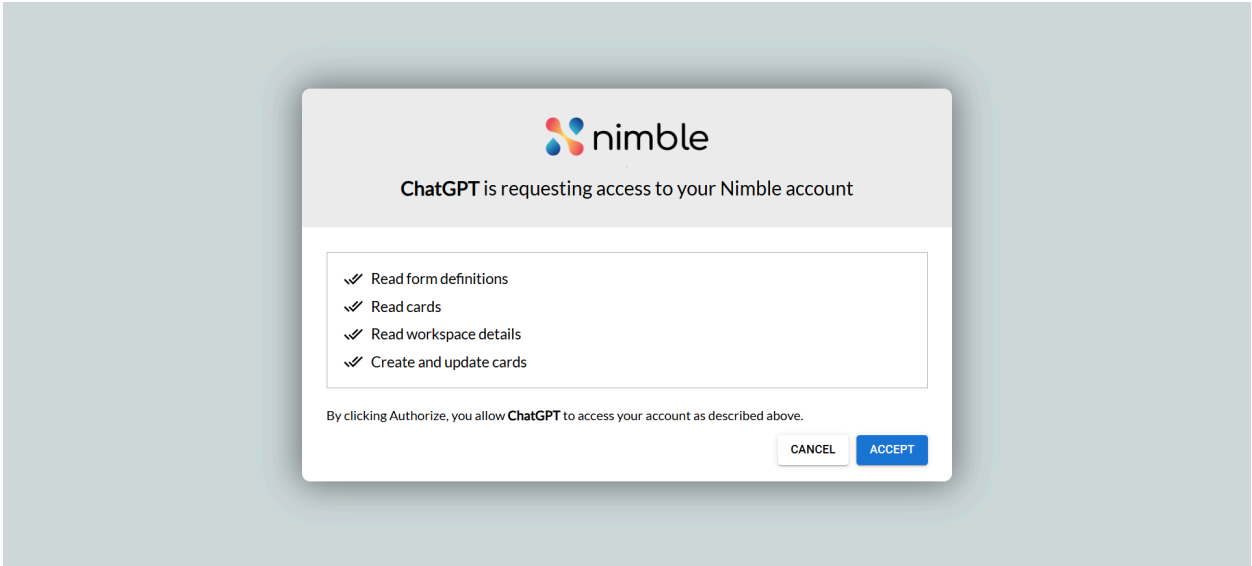
Field	Meaning
url	The NimbleWork MCP endpoint.
client_id	The OAuth client ID (provided by NimbleWork; it is not a secret).
callback_url / callback_port	The fixed local address Codex uses to receive the sign-in response. Use these exact values so they match what NimbleWork has authorized — do not change the port or path.

Step 3 — Authorize the connector

Connect the server to your Nimble account:

```
codex mcp login nimblework-mcp
```

A sign-in URL appears and your browser opens. Sign in with your Nimble account and click **ACCEPT** on the authorization screen.



When it finishes, the terminal shows :

```
Successfully logged in to MCP server 'nimblework-mcp'.
```

Verify the server is registered and authorized :

```
codex mcp list
```

You should see nimblework-mcp with Status = enabled and Auth = OAuth .

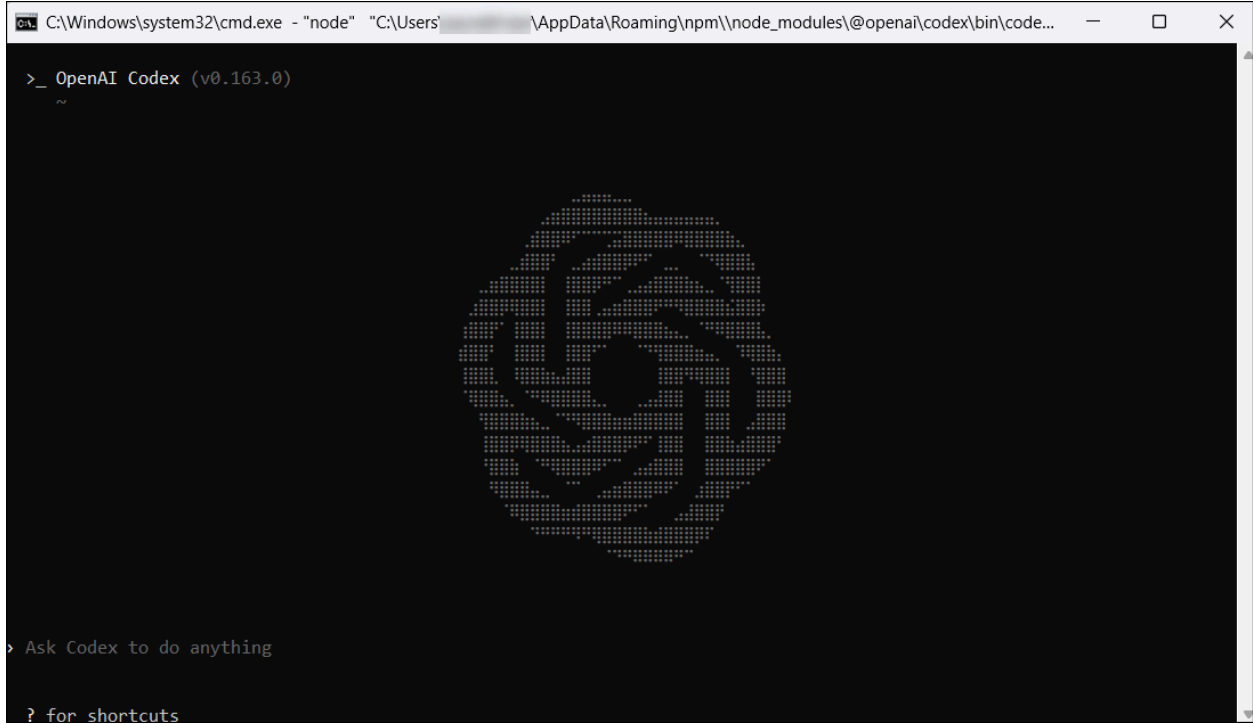
```
C:\Users\>codex mcp list
```

Name	Url	Bearer Token	Env Var	Status	Auth
nimblework-mcp	https://api.digite.com/mcp/v1	-		enabled	OAuth

Step 4 — Use it

Start Codex:

codex



```
saaurabh.kar

>_ OpenAI Codex (v0.163.0)
~

To get started, describe a task or try one of these commands:

/init - create an AGENTS.md file with instructions for Codex
/status - show current session configuration
/permissions - choose what Codex is allowed to do
/model - choose what model and reasoning effort to use
/review - review any changes and find issues

Set up the Codex agent sandbox to protect your files and control network access. Learn more
<https://developers.openai.com/codex/windows>

> 1. Set up default sandbox (requires Administrator permissions)
2. Use non-admin sandbox (higher risk if prompt injected)
3. Quit
enter select - esc back
```

Note: On first run, Codex asks you to set up its sandbox (to protect your local files and network access). Choose "Set up default sandbox" if you have Administrator rights, otherwise "Use non-admin sandbox", then continue. This is a one-time Codex setup and does not affect the NimbleWork connection.

```
saaurabh.kar

>_ OpenAI Codex (v0.163.0)
~

To get started, describe a task or try one of these commands:

/init - create an AGENTS.md file with instructions for Codex
/status - show current session configuration
/permissions - choose what Codex is allowed to do
/model - choose what model and reasoning effort to use
/review - review any changes and find issues

* Sandbox ready
Codex can now safely edit files and execute commands in your computer

> Ask Codex to do anything

GPT-6.1-Sol default ~
← for agents - ? for shortcuts
```

Then ask, in plain English, something that uses your Nimble data — for example:

List all my Nimble workspaces.

When Codex calls the NimbleWork connector, approve the tool call. Codex returns the result from your

Nimble account.

Tip: To run a single request without opening the interactive session, use: `codex exec --approve-for-me "List all my Nimble workspaces"`

Step 5 — Troubleshooting

Problem	Fix
403 Forbidden on the sign-in page	The callback in your config does not match what NimbleWork authorized. Use the exact url, client_id, callback_url, and callback_port shown in Step 2.
codex mcp login shows a URL but nothing opens	Copy the URL from the terminal into your browser manually, then sign in and ACCEPT.
"MCP tool call requires approval" when using codex exec	Headless runs block tool calls by default. Add --approve-for-me, or run interactively with codex and approve the call.
"Reading additional input from stdin..." seems stuck	codex exec is waiting for input. Provide your prompt as an argument, or pipe empty input so it proceeds.
TLS / certificate errors (corporate network)	Set your corporate root certificate before running Codex: set NODE_EXTRA_CA_CERTS=C:\path\to\your-cert.pem
codex: command not found	The Codex CLI is not installed or not on PATH. Install it, then reopen Command Prompt.

4. Support

For help with the connector, contact: mcp-support@nimblework.com.



NimbleWork MCP Connector