

# DevProxy

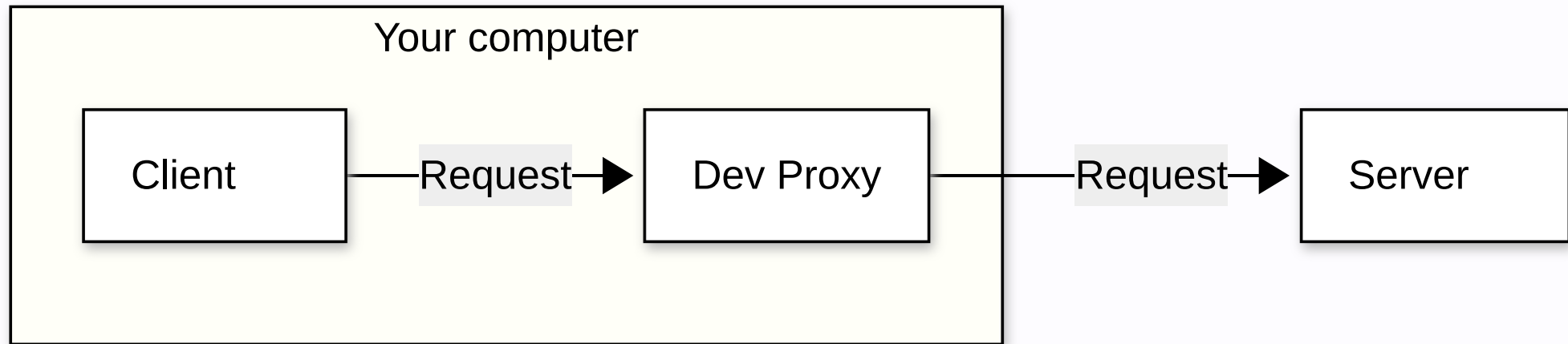
Dev Proxy is an API simulator that helps you effortlessly test your app beyond the happy path.

# Story

September 2025 CloudFlare outage on their Cloudflare Dashboard due to repeated calls to their own API - the cause was a useEffect without a dependency.

# What is Dev Proxy?

Dev Proxy is an API simulator that helps you effortlessly test your app beyond the happy path.



What is a proxy?

# Use Cases



- **Simulate API errors** - on internal (ie. localhost) and external (ie. Microsoft Graph).
- **Add latency** to requests to so you can test your UI display loading data messages.
- Fire **rate limits** and **handle throttling** - test with the `Retry-After` header.

- Stand up **mock APIs and data** - for when the backend is not ready yet
- **Intercept OpenAI-compatible requests** to **analyze costs**
- Check if your API is making requests with **least permissions** with **API Center**

# Get Started



# Local Development

-  Dev Proxy will set itself as the system proxy. You can run commands `Invoke-WebRequest` or `curl`.
-  Chromium\* based browsers (Edge/Chrome) bypass system proxy settings for localhost URLs - you need to exclude localhost URLs from the bypass list.

<sub>\*See [documentation](#) for Firefox.</sub>

⚠ Save your work beforehand ⚠  
Run this to test localhost

```
taskkill /f /im msedge.exe  
cd "C:\Program Files (x86)\Microsoft\Edge\Application"  
msedge --proxy-bypass-list="<-loopback>" --proxy-server="127.0.0.1:8000"
```



# Usage

```
req GET http://localhost:3000/meals
time 3/10/2025 1:45:24 AM +00:00
skip RetryAfterPlugin: Request not throttled
skip GenericRandomErrorPlugin: Pass through
oops LatencyPlugin: Delaying request for 420ms
pass Passed through
```

<sub>How does your application handle slow responses, rate limits and errors?</sub>

# Dev Proxy Toolkit

VSCode extension is a  
wrapper to the CLI.



# Config

Open the VSCode command palette `CTRL+SHIFT+P` and 'Dev Proxy Toolkit: Create create configuration file'

Creates `.devproxy/devproxyc.json`

```
{
  "$schema": "https://raw.githubusercontent.com/dotnet/dev-proxy/main/schemas/v1.0.0/rc.schema.json",
  "plugins": [
    {
      "name": "RetryAfterPlugin",
      "enabled": true,
      "pluginPath": "~appFolder/plugins/DevProxy.Plugins.dll"
    },
    {
      "name": "GenericRandomErrorPlugin",
      "enabled": true,
      "pluginPath": "~appFolder/plugins/DevProxy.Plugins.dll",
      "configSection": "genericRandomErrorPlugin"
    }
  ],
  "urlsToWatch": [
    "https://jsonplaceholder.typicode.com/*"
  ],
  "genericRandomErrorPlugin": {
    "errorsFile": "devproxy-errors.json"
  },
  "rate": 50,
  "logLevel": "information",
  "newVersionNotifiction": "stable",
  "showSkipMessages": true,
  "showTimestamps": true
}
```

# Plugins

💡 Hopefully this gives you some ideas 💡

- `Auth` - Simulates authentication and authorization using API keys or OAuth2.
- `CachingGuidancePlugin` - Shows a warning when Dev Proxy intercepted the same request within the specified period of time.
- `CrudApiPlugin` - Simulates a CRUD API with an in-memory data store.

- `EntraMockResponsePlugin` - Mocking auth flow API requests.
- `ExecutionSummaryPlugin` - Creates a summary of the requests that pass through the proxy.
- `MockGeneratorPlugin` - Generate mocks from the request.

- `MockRequestPlugin` - test webhooks in your client from dev proxy.
- `RateLimitingPlugin` - Simulates rate-limit behaviors.
- `RewritePlugin` - use rewrite rules - test example.com vs example.local
- `UrlDiscoveryPlugin` - creates a list of requested URLs.
- ... and much more

# Focus Mock and CRUD Plugin

*Return fixed data when your backend is not ready*

- Different data on `nth` request
- Can deliver binary data
- Create a mock CRUD API - mutates in memory and resets on restart.
- Access via `devtunnel` over the internet - have your cloud apps connect to your machine.
- Supports Microsoft Entra



# Next Steps

 View Online [neave.dev/devproxy](https://neave.dev/devproxy)

 Download [PowerPoint](#) and [PDF](#)

 Contribute on  [GitHub](#)