



macOS companion app > **Gateway on macOS**

macOS companion app

## Gateway on macOS

OpenClaw.app no longer bundles Node/Bun or the Gateway runtime. The macOS app expects an **external openclaw** CLI install, does not spawn the Gateway as a child process, and manages a per-user launchd service to keep the Gateway running (or attaches to an existing local Gateway if one is already running).

### Install the CLI (required for local mode)

You need Node 22+ on the Mac, then install `openclaw` globally:

```
npm install -g openclaw@<version>
```

The macOS app's **Install CLI** button runs the same flow via npm/pnpm (bun not recommended for Gateway runtime).

### Launchd (Gateway as LaunchAgent)

Label:

`bot.molt.gateway` (or `bot.molt.<profile>` ; legacy `com.openclaw.*` may remain)

Plist location (per-user):



```
~/Library/LaunchAgents/bot.molt.gateway.plist (or  
~/Library/LaunchAgents/bot.molt.<profile>.plist )
```

Manager: >

The macOS app owns LaunchAgent install/update in Local mode.

The CLI can also install it: `openclaw gateway install .`

Behavior:

“OpenClaw Active” enables/disables the LaunchAgent.

App quit does **not** stop the gateway (launchd keeps it alive).

If a Gateway is already running on the configured port, the app attaches to it instead of starting a new one.

Logging:

```
launchd stdout/err: /tmp/openclaw/openclaw-gateway.log
```

## Version compatibility

The macOS app checks the gateway version against its own version. If they’re incompatible, update the global CLI to match the app version.

## Smoke check

```
openclaw --version  
  
OPENCLAW_SKIP_CHANNELS=1 \  
OPENCLAW_SKIP_CANVAS_HOST=1 \  
openclaw gateway --port 18999 --bind loopback
```

Then:

```
openclaw gateway call health --url ws://127.0.0.1:18999 --timeout 3000
```

&gt;

[macOS Release](#)[macOS IPC](#) >Powered by [mintlify](#)