Module 02

Preparing Your Laptop

🦞 USB Setup Kit, WSL2, and Everything You Need

OpenClaw Course

Navigation Chart

By the end of this module, you will be able to:

1. **Verify** your laptop meets the minimum requirements
2. **Explain** why WSL2 is required and what it does
3. **Prepare** a USB Setup Kit with WSL installer + Ubuntu image
4. **Install** WSL2 and Ubuntu from USB — no Store, no network needed
5. **Enable** systemd inside WSL2 (required for the OpenClaw daemon)
6. **Install** Node.js 22+ inside your WSL2 environment
7. **Verify** your entire environment is ready for OpenClaw
8. **Configure** your laptop for 24/7 operation

📜 Ship's Logbook (Part 1)

Term	Definition
WSL2	Windows Subsystem for Linux v2 -- run a full Linux OS inside Windows
Ubuntu	A beginner-friendly Linux distribution we install inside WSL2
Terminal	Text-based interface for typing commands
systemd	Linux service manager -- OpenClaw's daemon needs it to start automatically
Daemon	A program that runs continuously in the background
Node.js	JavaScript runtime that 🦞 OpenClaw is built on (version 22+ required)

📜 Ship's Logbook (Part 2)

Term	Definition
npm	Node Package Manager -- installs JavaScript packages
nvm	Node Version Manager -- install and switch between Node.js versions
USB Setup Kit	A USB stick with the WSL installer and Ubuntu image, prepared on your main computer
PATH	List of directories your computer checks when you type a command
Tailscale	Secure networking tool for private remote access between your devices
Loopback	Network address (127.0.0.1) that only your own computer can reach

Used or Pre-Owned Laptops

"Reset this PC" does NOT remove corporate policies. Admin restrictions, MDM enrollment, and monitoring software survive a factory reset.

Check for corporate remnants

- **Start > System** — is a domain listed? (should say "WORKGROUP")
- **Settings > Accounts > Access work or school** — any organizations?
- Can you open PowerShell as admin? (`Ctrl + Shift + Enter`)

If restricted → clean install from USB (not a reset). Download the Media Creation Tool from microsoft.com, boot from USB, choose "Custom Install," and wipe all partitions.

> *Without admin access, WSL2 installation will be blocked.*

Check Your Hardware: RAM

Open **Task Manager** (`Ctrl + Shift + Esc`) > **Performance** tab > **Memory**.

RAM	Verdict
4 GB or less	Not enough -- will struggle
8 GB	Minimum -- works but slow when multitasking
16 GB	Excellent -- plenty for OpenClaw
32 GB+	Overkill, but great for local models later

What uses the RAM: WSL2 uses 1-4 GB, Node.js a few hundred MB. The rest is yours.

Check Your Hardware: Disk Space

Open **File Explorer** > **This PC** > Check your C: drive.

Free Space	Verdict
Less than 10 GB	Not enough
10-20 GB	Tight but workable
20-50 GB	Comfortable
50 GB+	Plenty of room

Total footprint: WSL2 + Ubuntu (~5 GB) + Node.js (~200 MB) + OpenClaw (~500 MB) + agent data over time (1-10 GB). About **20 GB** covers everything comfortably.

Check Your Hardware: Windows Version

Press `Win + R`, type `winver`, press Enter.

Version	WSL2 Support
Windows 10 v1903+ (Build 18362)	Supported
Windows 10 v2004+	Supported with simplified setup
Windows 10 22H2	Current -- fully supported
Windows 11	Fully supported

> ***If your version is older than 1903:*** *Update Windows first. Go to* ***Settings > Update & Security > Windows Update*** *and install all available updates.*

Why WSL2 Is Required

🦞 **OpenClaw is built for Linux/macOS.** The ⛵ gateway, daemon, 🐚 shell commands, and file permissions all assume a Unix-like OS.

> *"OpenClaw on Windows is recommended via WSL2 (Ubuntu recommended)." -- Official docs*

What WSL2 Actually Is

- Runs a **real Linux kernel** inside Windows
- Not an emulator, not a traditional VM
- Runs **alongside** Windows, not instead of it
- Access Windows files from Linux, and vice versa

Apartment analogy: Your laptop is a building with two tenants -- Windows and Linux. They share the hardware but have their own separate units.

Why an Old Laptop?

Most tutorials say: rent a VPS or buy a Mac Mini. Here's why a **used Windows laptop** wins:

Option	Cost	Privacy	Control
Old laptop	$0-100	Data stays home	You own everything
VPS	$5-20/mo forever	Provider's computer	Provider can access it
Mac Mini	$500-800	Good, but pricey	Overkill for one agent

The old laptop wins on cost, privacy, AND control. You pay once (or $0 if you already have one), your data never leaves your house, and nobody else can touch it.

Avoid Plug-and-Play Services

Some companies sell pre-configured OpenClaw boxes or hosted setups. **Be very careful.**

> *"DO NOT use a service that sets up OpenClaw for you. I tried 5 popular services and pretty much all of them exposed the gateway, didn't have pairing mode, and allowed the root directory to be discovered by the internet."*

What "exposed the gateway" means

- Anyone on the internet could connect to the agent
- No pairing mode = no verification that YOU are the owner
- Root directory exposed = your files visible to the world

This course teaches you to set it up yourself — correctly and securely.

Step 1: Connect to Guest WiFi & Update Windows

Before installing anything, the laptop needs to be updated. But it should **never** connect to your main home WiFi.

Set up a guest network (from your phone or main computer)

1. Find your router address: run `ipconfig`, note the **Default Gateway**
2. Open that address in a browser to reach router admin
3. Enable **Guest Network** with a password, disable "access local network"

Connect the OpenClaw laptop to guest WiFi, then update

1. Connect to the **guest** network only
2. **Settings > Update & Security > Windows Update**
3. Click **"Check for updates"** — install everything, restart when prompted
4. **Expect 20-30 minutes** — check for updates again after each restart

Step 2: Prepare Your USB Setup Kit

This step happens on your **main computer** (not the OpenClaw laptop). You need a USB stick with at least 1 GB free.

Download three files and copy all to your USB stick

1. **WSL installer** (.msi) — from **github.com/microsoft/WSL/releases**, download the **x64** `.msi` (~70 MB)
2. **Ubuntu image** — from **cloud-images.ubuntu.com/wsl/releases/24.04/current/**, download the **amd64** `.rootfs.tar.gz` (~340 MB)
3. **Setup commands** — `setup-commands.txt` from the course materials (every command you need, ready to copy and paste)

> *If the Ubuntu link is empty, navigate from **cloud-images.ubuntu.com/wsl/** → releases → 24.04 → current*

Step 3: Install WSL from USB

Plug the USB into the **OpenClaw laptop**. No WiFi needed.

1. **Copy all files** from USB to your **Downloads** folder
2. In Downloads, **double-click** the `.msi` file — click **Yes**. Installs silently — that's normal.
3. Open **PowerShell as Admin** (Start > type "PowerShell" > `Ctrl + Shift + Enter`)
4. Open `setup-commands.txt` from Downloads. Copy-paste **STEP 1**:

```
& 'C:\Program Files\WSL\wsl.exe' --install --no-distribution
```

5. **Restart** your computer (Start > Power > Restart)

> *The .msi installs the WSL program. The command enables the Windows features WSL2 needs. Both are required.*

Step 4: Import Ubuntu

After restarting, open **PowerShell as Administrator** (`Ctrl + Shift + Enter`).

Paste **STEP 2** and **STEP 3** from `setup-commands.txt`:

```
mkdir C:\WSL\Ubuntu
& 'C:\Program Files\WSL\wsl.exe' --import Ubuntu C:\WSL\Ubuntu "$env:USERPROFILE\Downloads\ubuntu-noble-wsl-amd64-wsl.rootfs.tar.gz"
```

No output = success. Verify with **STEP 4**:

```
& 'C:\Program Files\WSL\wsl.exe' -l -v
```

Expected: **Ubuntu, VERSION 2**

> `$env:USERPROFILE` *expands to your user folder automatically. No drive letters to guess — that's why we copied to Downloads.*

Step 5: Configure Ubuntu (Create User)

Open Ubuntu and create your account:

```
& 'C:\Program Files\WSL\wsl.exe' -d Ubuntu
```

You'll see a `root@` prompt. **To paste in Ubuntu: right-click** (Ctrl+V does NOT work here — right-click pastes automatically). Run these one at a time:

```
adduser openclaw
usermod -aG sudo openclaw
echo -e "[boot]\nsystemd=true\n\n[user]\ndefault=openclaw" > /etc/wsl.conf
```

- **Password won't show** when you type it -- no dots, no stars, nothing. This is normal!
- Skip optional fields (Full Name, Room, etc.) by pressing Enter

Type `exit`, then in PowerShell: `& 'C:\Program Files\WSL\wsl.exe' --shutdown`. Wait 5

Step 5: Configure Ubuntu (Start Menu + Verify)

Verify Start Menu shortcut

Type **ubuntu** in the Start menu. If it appears and launches, you're set — skip ahead.

If it doesn't appear, create one manually:

1. `Win+R` → type `shell:programs` → Enter
2. Right-click > **New** > **Shortcut**
3. Location: `"C:\Program Files\WSL\wsl.exe" -d Ubuntu`
4. Name it **Ubuntu**, click Finish

Verify systemd

Open Ubuntu from the Start Menu and run:

```
systemctl is-system-running
```

Step 7: Update Ubuntu and Install Tools

In your Ubuntu terminal:

Update packages (5-10 minutes on first run)

```
sudo apt update && sudo apt upgrade -y
```

Install essential tools

```
sudo apt install -y git curl wget build-essential
```

These provide version control, download utilities, and compilation tools needed by npm.

Step 8a: Install nvm

We use **nvm** (Node Version Manager) -- the recommended way.

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.1/install.sh | bash
```

Then activate it:

```
source ~/.bashrc
nvm --version           # Should show: 0.40.1
```

> *If you see `nvm: command not found` -- close the Ubuntu window, then reopen it: **Start Menu** (type "Ubuntu") or **PowerShell**: `& 'C:\Program Files\WSL\wsl.exe' -d Ubuntu`*

Step 8b: Install Node.js 22

> *If paste isn't working, close the Ubuntu window, then reopen it:* ***Start Menu*** *(type "Ubuntu") or* ***PowerShell****:* `& 'C:\Program Files\WSL\wsl.exe' -d Ubuntu`

```
nvm install 22
nvm alias default 22
```

Verify both

```
node --version          # Should show: v22.x.x
npm --version           # Should show: 10.x.x
```

As long as `node` starts with v22, you're fueled up.

Step 9a: Power Settings — Disable Sleep

For 24/7 operation, your laptop must stay awake. Two settings must BOTH be changed.

Idle sleep timeout

- Settings > System > Power & sleep → set sleep to Never

Lid close behavior (most guides miss this!)

1. Control Panel > Hardware and Sound > Power Options
2. Click Choose what closing the lid does
3. Set "When I close the lid" → Do nothing (both columns)
4. Click Save changes

> *Without BOTH settings, your laptop sleeps when the lid closes — regardless of idle timeout.*

Step 9b: Hibernate, Display, and Plug In

Disable hibernate (requires admin PowerShell)

```
powercfg /hibernate off
```

> *Start menu > type "PowerShell" > press* `Ctrl + Shift + Enter` *for admin. A regular terminal will reject this command.*

Display can turn off — that's fine

- "Turn off screen after" → **5 minutes** (display off, computer still runs)

Keep the laptop **plugged in** — running 24/7 on battery is not practical.

Step 9c: Power Settings — Verify

Test with the lid closed (not just screen lock)

1. Close your laptop lid
2. Wait one minute
3. Open the lid, log back in
4. Open your **Ubuntu terminal** (not PowerShell) and run:

```
uptime
```

You should see **continuous uptime** — not a fresh boot. If it restarted:

- Recheck **lid close** in Control Panel (Step 9a)
- Recheck **sleep** in Settings (Step 9a)

Step 10: Understand the File System Boundary

There are **two separate file systems**. Knowing which you are in matters.

Location	Path Style	Example
Windows	`C:\Users\...`	`C:\Users\GC\Documents\`
Linux (WSL2)	`/home/username/...`	`/home/openclaw/`

The Golden Rule

> *OpenClaw's data should live in the Linux file system, NOT the Windows file system.*

Files in `/home/openclaw/` are fast. Files through `/mnt/c/` are **significantly slower** (every operation crosses the WSL2 boundary). Run `pwd` to check which side you're on.

Optional: Tailscale (Remote Access)

Tailscale lets you access OpenClaw from your phone or another computer. **Skip if** you'll only use this laptop locally.

```
curl -fsSL https://tailscale.com/install.sh | sh
sudo tailscale up
```

This gives you a URL to open in your browser to authorize the device. Once connected:

- **From this laptop:** `http://127.0.0.1:18789/`
- **From your phone (on Tailscale):** `http://100.x.x.x:18789/`
- **From everywhere else:** no access (secure by default)

Optional: Disk Encryption

If your laptop stores API keys, conversation history, and agent memory — encrypt the disk.

Platform	How
Windows Pro/Enterprise	BitLocker (Settings > Update & Security)
Windows Home	Device Encryption (requires Microsoft account)

> *If that laptop walks off, encrypted data is unreadable without your password.*

Shoals and Sandbars (1/2)

Problem	Cause	Fix
"Warning 1964" during .msi install	Cosmetic shortcut property	Ignore — WSL installed fine
Error `0x80370102`	Virtualization disabled in BIOS	Enable "Intel VT" or "AMD-V" in BIOS settings
"WSL optional component not enabled"	Windows features not activated	Run `--install --no-distribution` from Step 3 and restart
USB stick not detected	Bad port or drive letter	Try another USB port; check File Explorer for drive letter
`wsl --import` error	Wrong file path	Paste command from `setup-commands.txt` (path is pre-filled)

Shoals and Sandbars (2/2)

Problem	Cause	Fix
Paste stops working	Terminal gets stuck	Close the Ubuntu window, reopen from Start Menu
`node: command not found`	nvm not loaded	Run `source ~/.bashrc` or reopen from Start Menu
Password not accepted	Using Windows password	Use the Linux password from Ubuntu setup
Laptop sleeps when lid closes	Lid close not configured	Control Panel > Power Options > lid → Do nothing
"Run as admin" unavailable	Corporate policy remnants	Clean install from USB (not a reset)

The Verification Checklist

Run each command in your 🐚 Ubuntu terminal:

Check	Command	Expected
WSL2 version	`wsl.exe -l -v`	Ubuntu, VERSION 2
systemd	`systemctl is-system-running`	`running` or `degraded`
Node.js	`node --version`	`v22.x.x`
npm	`npm --version`	`10.x.x`
Git	`git --version`	`git version 2.x.x`
curl	`curl --version`	Any version output
File system	`pwd`	`/home/openclaw` (not `/mnt/c/`)

All seven must pass before moving to Module 03.

⚙ Hands on Deck

Complete this checklist:

- [ ] **Hardware:** 8+ GB RAM, 20+ GB free disk, Windows 1903+
- [ ] **USB Setup Kit:** Prepared with WSL .msi + Ubuntu .tar.gz
- [ ] **WSL2:** Installed from USB, VERSION 2, username created
- [ ] **systemd:** Enabled and running
- [ ] **Node.js + npm:** v22.x.x and 10.x.x installed
- [ ] **Git:** Installed
- [ ] **Power settings:** Sleep disabled AND lid close set to "Do nothing"
- [ ] **Network:** Connected to guest WiFi only
- [ ] **Bonus:** Navigate to `\\wsl$\Ubuntu\home\openclaw\` in File Explorer

💎 Treasure Chest

1. **Prepare a USB Setup Kit** -- WSL .msi + Ubuntu image + setup-commands.txt on your main computer
2. **Copy to Downloads, then install** -- no drive letters to guess, commands are ready to paste
3. 🦞 **OpenClaw requires WSL2** -- it runs inside Linux, not directly on Windows
4. **systemd is enabled** during Ubuntu configuration -- one command handles it
5. **Node.js 22+ is required** -- install with nvm for easy version management
6. **Keep 🦞 OpenClaw files in the Linux file system** -- `/home/username/`, not `/mnt/c/`
7. **Disable sleep AND configure lid close** -- both Settings and Control Panel must be changed
8. **Guest WiFi only** -- the OpenClaw laptop never connects to your main home network

🌊 Next Port of Call

Module 03: Installing 🦞 OpenClaw

> *Your laptop is ready — installed from USB, configured, and verified. Time to run the installer, walk through the onboarding wizard, and bring your agent to life.*