readme.md for @ecobridge.xyz/eco_os

? EcoOS

> **A purpose-built, minimal Linux distribution for kiosk and digital signage deployments.**

EcoOS is a streamlined operating system that boots directly into a full-screen Chromium browser, managed by a powerful daemon with a built-in web UI. Perfect for digital signage, interactive kiosks, info displays, and any scenario where you need a locked-down, browser-based interface.

Issue Reporting and Security

For reporting bugs, issues, or security vulnerabilities, please visit community.foss.global/. This is the central community hub for all issue reporting. Developers who sign and comply with our contribution agreement and go through identification can also get a code.foss.global/ account to submit Pull Requests directly.

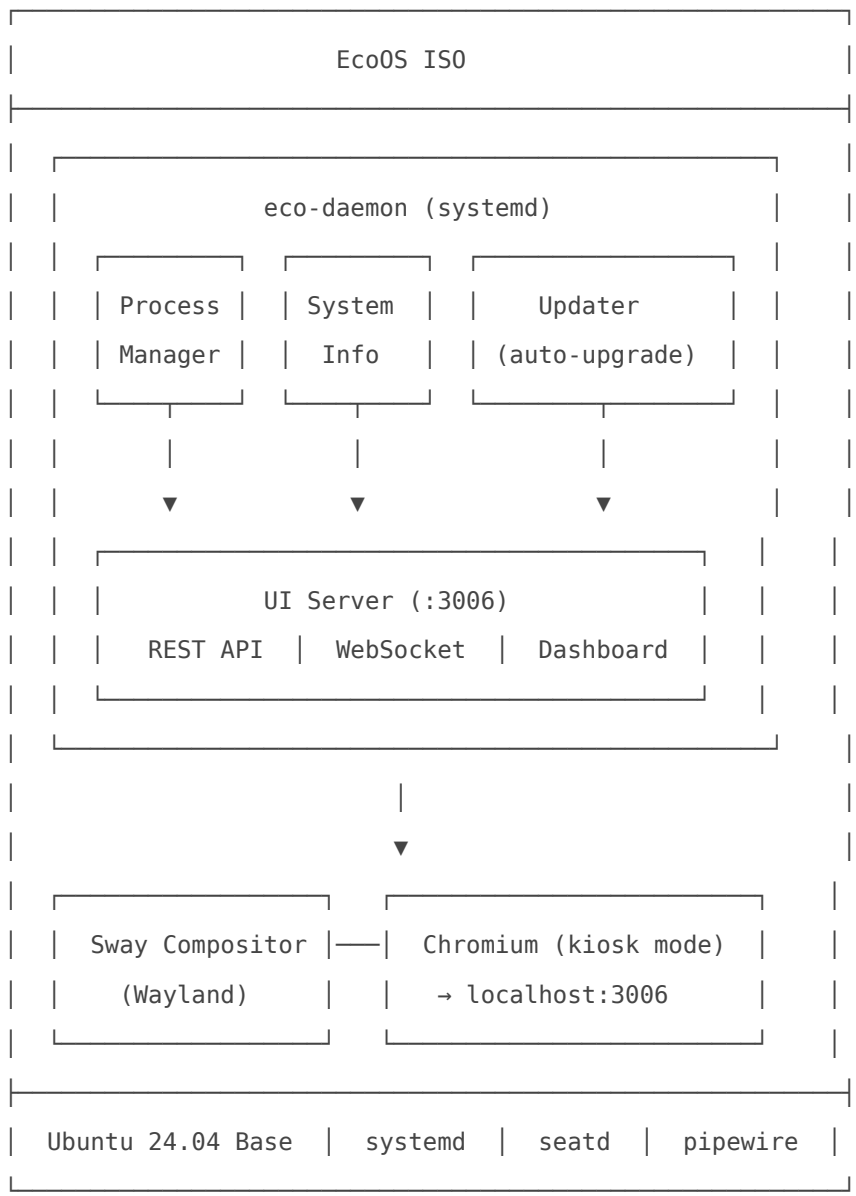
? Features

- □□ **Zero-Config Boot** — Boots straight into a Wayland-based kiosk browser
- □□ **Sway Compositor** — Modern, tiling Wayland compositor with automatic fallback modes
- □□ **Chromium Kiosk** — Full-screen browser in locked-down kiosk mode
- □□ **Management UI** — Real-time system monitoring and control via web interface on port 3006
- □□ **Auto-Updates** — Daemon self-updates with smart stability checking
- □□ **System Monitoring** — CPU, memory, disk, network, GPU, and audio device stats
- □□ **Live Logs** — System journal and daemon logs accessible from the UI
- □□ **Hardware Support** — Input devices, speakers, microphones detection and display

- ⚡ **Rapid Recovery** — Auto-restart of crashed services within seconds

---

?? Architecture



?? Development

Prerequisites

- **Docker** (for ISO building)

- **pnpm** (package manager)
- **Deno** v2.x (for daemon development)
- **QEMU** (for testing)

Project Structure

```
eco_os/
├── ecoos_daemon/          # Daemon source (Deno/TypeScript)
│   ├── mod.ts             # Entry point
│   └── ts/
│       ├── daemon/        # Core daemon logic
│       │   ├── index.ts       # EcoDaemon class
│       │   ├── process-manager.ts  # Sway/Chromium management
│       │   ├── system-info.ts # Hardware detection
│       │   └── updater.ts     # Auto-update system
│       ├── ui/            # Web UI server
│       └── utils/         # Utilities
├── isobuild/              # ISO build configuration
│   ├── Dockerfile         # Build container
│   ├── config/            # live-build config
│   └── scripts/           # Build scripts
├── isotest/               # QEMU test scripts
└── .nogit/                # Generated artifacts (not in git)
    ├── iso/               # Built ISO
    ├── vm/                # QEMU files
    └── screenshots/       # VM screenshots
```

Commands

```
# Build the full ISO (auto-rebuilds daemon first)
pnpm run build

# Test ISO in QEMU virtual machine
pnpm run test

# Take screenshot of running VM
pnpm run test:screenshot

# Stop the QEMU VM
```

```
pnpm run test:stop

# Clean all build artifacts
pnpm run clean

# Daemon development (watch mode)
pnpm run daemon:dev

# Bundle daemon to standalone binary
pnpm run daemon:bundle

# Type-check daemon code
pnpm run daemon:typecheck
```

?? Management UI

The daemon exposes a management interface at `http://localhost:3006` (or the device's IP on port 3006).

Dashboard Features

Panel	Description
Services	Status of Sway compositor and Chromium browser
CPU	Model, core count, real-time usage
Memory	Used/total with visual progress bar
Network	Interface names and IP addresses
Disks	Mount points, usage, and capacity
System	Hostname, uptime, GPU info
Controls	Restart browser, reboot system buttons
Updates	Version info, available updates, upgrade controls
Input Devices	Keyboards, mice, touchscreens
Audio	Detected speakers and microphones
Logs	Daemon logs and system journal viewer

API Endpoints

Endpoint	Method	Description
`/api/status`	GET	Full system status
`/api/logs`	GET	Daemon logs
`/api/reboot`	POST	Reboot the system
`/api/restart-chromium`	POST	Restart the kiosk browser
`/api/updates`	GET	Update information
`/api/updates/check`	POST	Check for new updates
`/api/upgrade`	POST	Upgrade to specific version
`/ws`	WebSocket	Real-time status updates

? Update System

EcoOS features a smart auto-update mechanism:

1. **Hourly Checks** — Daemon polls for new releases every hour
2. **Stability Period** — New releases wait 24 hours before auto-upgrade (prevents deploying unstable releases)
3. **Seamless Upgrade** — Downloads new daemon binary, replaces, and restarts service
4. **Manual Override** — Force immediate upgrade via UI or API
5. **Version Tracking** — UI auto-reloads when daemon version changes

---

? Testing

QEMU Virtual Machine

```
# Start VM (creates disk, boots ISO)
pnpm run test

# Take screenshots to monitor progress
pnpm run test:screenshot

# Screenshot loop (every 5 seconds)
pnpm run test:screenshot:loop
```

```
# Stop VM
pnpm run test:stop

# Clean and restart fresh
pnpm run test:clean && pnpm run test
```

Serial Console

For debugging without graphics:

```
socat - UNIX-CONNECT:.nogit/vm/serial.sock
# Login: ecouser / ecouser
# Root: sudo -i
```

? Release Assets

Each release includes:

File	Description
`ecoos-vX.X.X.iso`	Full bootable ISO image (~2GB)
`eco-daemon-vX.X.X`	Standalone daemon binary for in-place upgrades
`SHA256SUMS.txt`	Checksums for verification

? Boot Menu Options

1. **Install EcoOS** *(default, auto-selects in 10s)* — Full installation to disk
2. **EcoOS Live** — Try without installing (runs from RAM)
3. **EcoOS Live (Safe Mode)** — Minimal boot for troubleshooting

---

?? Technical Details

Daemon

- **Runtime**: Deno (compiled to standalone binary)
- **Process Management**: Spawns and monitors Sway + Chromium
- **Backend Fallback**: Tries DRM first, falls back to headless/pixman
- **Auto-Recovery**: Restarts crashed services within 5 seconds
- **Logging**: Integrates with systemd journal

Kiosk Browser

- **Browser**: Chromium (official snapshots, not snap)
- **Flags**: `--ozone-platform=wayland --kiosk --no-first-run --disable-infobars`
- **Default URL**: `http://localhost:3006` (management UI)

System Stack

- **Base**: Ubuntu 24.04 LTS
- **Init**: systemd
- **Display**: Sway (Wayland compositor)
- **Seat Manager**: seatd
- **Audio**: PipeWire

License and Legal Information

Trademarks

Company Information

Task Venture Capital GmbH Registered at District Court Bremen HRB 35230 HB, Germany

For any legal inquiries or further information, please contact us via email at hello@task.vc.

By using this repository, you acknowledge that you have read this section, agree to comply with its terms, and understand that the licensing of the code does not imply endorsement by Task Venture Capital GmbH of any derivative works.

Revision #6
Created 2026-03-28 10:49:15 UTC by foss.global Team
Updated 2026-03-28 12:14:40 UTC by foss.global Team