# FaceForge Desktop (Sprint 12)

This folder contains the **Desktop shell MVP** (Tauri v2) that makes FaceForge feel like a real app:

- First-run wizard to choose `FACEFORGE_HOME` and ports
- Local process orchestration (Core, optional SeaweedFS)
- Tray UX (close-to-tray, Open UI, Status, Logs, Stop/Restart, Exit)

## Dev run (Windows)

Prereqs:

- Rust toolchain (stable)

From the repo root:

- `cargo tauri dev --manifest-path .\desktop\src-tauri\Cargo.toml`

Notes:

- Desktop launches Core using the repo-local `.venv\\Scripts\\python.exe` if present.
- Desktop sets `PYTHONPATH` to `core\\src` so Core can run without `pip install -e`.

## Settings + files

On first run, Desktop asks for:

- `FACEFORGE_HOME` (data directory)
- Core port (default suggestion: 43210)
- Optional SeaweedFS S3 port (default suggestion: 43211)

Desktop writes:

- `${FACEFORGE_HOME}/run/ports.json`
- `${FACEFORGE_HOME}/config/core.json` (including `auth.install_token`)

Desktop also stores a tiny bootstrap pointer in the OS app config directory so it can find the chosen `FACEFORGE_HOME` on subsequent runs.

## Token UX

Core requires a per-install token for non-health endpoints.

- Desktop displays the token and offers a copy button.
- "Open UI" opens `/ui/login` so you can paste the token once (it becomes a cookie).