⬡ QUICK START GUIDE

SYNLYNK

From zero to multi-agent workgroup in 5 minutes.

v0.38.3 · Measurement & Reliability

```
$ pipx install git+https://github.com/nikhilsoman/synlynk

✓ synlynk 0.12.0 installed to isolated venv
✓ PATH updated · zero pip dependencies

→ cd your-repo && synlynk init
```

🐍 Python 3.8+ | 🔌 Zero Dependencies | 🤖 4 Agents | ⚡ 1 Command

INSTALL & INIT

Install once.
Init in every repo.

Synlynk is one Python package with no pip dependencies. Install it globally, then run `synlynk init` in any git repo to bootstrap your docs and agent instruction files.

1 **Install globally**

Installs `synlynk` into an isolated pipx venv and puts the binary on your PATH. No sudo, no build.

install

```
$ pipx install git+https://github.com/nikhilsoman/synlynk
✓ Installed synlynk into isolated pipx venv
✓ Binary at ~/.local/bin/synlynk · PATH updated
✓ synlynk 0.12.0 ready
```

2 **Bootstrap the repo**

The init wizard asks for mode (single / team) and discovers your installed agents. It migrates any existing docs rather than overwriting them.

synlynk init

```
~/my-project $ synlynk init
Mode? [single / team] › single
Scanning repository for existing docs...
✓ project-docs/roadmap.md · memory.md · costs.md
✓ project-docs/todo.md (generated from SQLite stories)
✓ project-docs/devlogs/
✓ CLAUDE.md · GEMINI.md · AGENTS.md · .cursorrules
✓ .synlynk/ (config.json · state.db · telemetry)
✓ 3 agents discovered claude · agy · codex
→ Ready. Try: synlynk exec claude
```

What gets created

📁

project-docs/

roadmap.md, memory.md, costs.md, per-user devlogs. `todo.md` is generated from the SQLite stories table — never hand-edited.

🤖

Agent instruction files

CLAUDE.md, GEMINI.md, AGENTS.md, .cursorrules — every agent reads the same project context from one init.

🗄️

.synlynk/

config.json (budget limits + docs dir), state.db (SQLite canon), telemetry.json (last 100 runs).

⚡

Nothing overwritten

Existing docs are migrated verbatim. Re-running init is safe — only missing files are written.

⚡ **SMART DOC MIGRATION**

Synlynk searches the target dir, repo root, `project-docs/`, and project-prefixed names. First file >200 bytes wins and is migrated as-is. Your work, not a blank skeleton.

FIRST RUN & DISPATCH

Four commands. That's the whole loop.

1 Run an agent with context

`synlynk exec claude` injects memory, roadmap, todos, and a source scan, then hands you an interactive session.

```
exec
$ synlynk exec claude
📋 Context injected (memory · roadmap · todos · scan) · 💰 $0.00 / $10.00
claude › Based on your roadmap, the next task is...
```

2 Dispatch a background job

`synlynk dispatch` fires an agent non-interactively and returns a job ID. Your terminal stays free.

```
dispatch
$ synlynk dispatch claude --task "Write unit tests for the billing module"
✓ Pre-flight gate passed · context-mode: task
✓ Dispatched job-a3f2 (claude) running in background
```

3 Check job status

`synlynk jobs` reads SQLite. Add `--watch` to follow live.

```
jobs
$ synlynk jobs
● job-a3f2 claude running 2m14s
```

4 Tail the output

`synlynk logs --job <id>` streams a job's captured log.

```
logs
$ synlynk logs --job job-a3f2
Writing test_billing.py...
→ 12/12 tests passing
```

🚀 ACHIEVEMENT UNLOCKED

Workgroup Commander

You dispatched an agent in the background, watched it run, and tailed its output — all from one CLI.

💡 **SCOPED CONTEXT SAVES TOKENS**

Add `--story <id>` or `--context-mode task` to a dispatch and synlynk injects only the story-scoped slice — typically ~400 bytes instead of 100KB of full context.

TEAM MODE

Onboard a teammate in one command.

When a new teammate clones a synlynk repo, `synlynk join` seeds their personal devlog from git history, regenerates their agent instruction files, and prints a team digest so they're caught up before their first session.

```
synlynk join

~/acme-api $ synlynk join
✓ Identified you as priya (git config user.name)
✓ Seeded devlogs/priya.md from 38 commits
✓ Regenerated CLAUDE.md · GEMINI.md · AGENTS.md
── Team digest ──────────────────────────
nikhil last active 2h ago · shipped relay broker
arjun last active 1d ago · capability routing
→ You're caught up. Try: synlynk team status
```

See where the team is

```
team status

$ synlynk team status
3 members · 12 open stories
nikhil 2 active jobs
priya onboarded today
arjun idle
```

Run a consensus decision

```
decide

$ synlynk decide --topic "DB?"
claude › Postgres
agy › Postgres
codex › Postgres
✓ Consensus: Postgres
```

THE TEAM CONTRACT

Every teammate gets their own devlog. Every agent reads the same context. Decisions in `memory.md` carry `[@username]` attribution — accountability without meetings.

STAY IN SYNC IN REAL TIME V0.9.4

Run `synlynk relay start` to open an HTTP SSE broker on port 27472, then `synlynk relay broadcast` to push a message, MOTD, or wellness check to the whole workgroup.

KEY FEATURES AT A GLANCE

Eight reasons teams adopt synlynk.

Everything below ships in a single `synlynk init` — no plugins, no config files to hand-write, no pip install.

🧠

Context Injection V0.1

Roadmap, memory, todos, and a source-architecture snapshot prepended before every agent session. No copy-pasting.

⚡

Background Dispatch V0.4

Fire agents as tracked background jobs. List them, watch them live, tail their logs — all from one CLI.

🎯

Capability Routing V0.5

A SQLite ledger scores each agent across engineering, org, and industry domains. The best agent for a story is picked automatically.

🛰️

Async Daemon V0.9.3

Run synlynk as an always-on launchd/systemd service with a job queue, priority chains, and crash-safe restarts.

💰

Budget Guard V0.1

A hard cost ceiling per session. Synlynk blocks execution the moment you'd cross your configured limit.

🛡️

Sentinel Alerts V0.1

Detects flatline loops, success loops, quota exhaustion, and skipped verification — before they cost you.

👥

Team Onboarding V0.9.2

`synlynk join` seeds a new member from git history and prints a team digest. Caught up before their first run.

📡

Relay Broadcast V0.9.4

An HTTP SSE broker connects your workgroup in real time. Push messages, MOTDs, and wellness checks to every member.

472	0	3	1
TESTS PASSING	PIP DEPENDENCIES	AGENTS OUT OF BOX	COMMAND TO START

📘 **GOING DEEPER**

This guide gets you running. For every command and flag see the **Command Reference**; for architecture, the feature matrix, and the full changelog see the **Official Reference**.