Multi-site RF–IP Gateway Platform

Linking analog radio sites over the operator's own servers

New radio sites join the network by pairing in the panel — no new code

CLIENT	Radio network operator Anonymous
INDUSTRY	Telecommunications
PROJECT TYPE	Custom Software
DURATION	2 months
TECHNOLOGIES	Php-Laravel, Python, Go, MySQL, Redis, Docker, WireGuard, Raspberry Pi

Running radio sites over the internet from one panel

A self-hosted EchoLink alternative built from Raspberry Pi gateways, a low-latency media relay and a Laravel panel.

Highlights

4	**20 ms**	**T01–T16**
Independent system planes	Opus audio frame	Defined acceptance tests

The problem: scattered sites, invisible faults

The usual way to link analog radio sites is to rely on a third-party internet linking service — leaving connection policies, monitoring data and device management outside the operator's control. The hard part is in the field, not in software: RF interference, internet outages, wrong audio levels or a stuck PTT are not fixed by a new app. The goal was to make these problems measurable and diagnosable remotely. Add a new site without writing code
Split repeaters into independent talk groups, or join them under control
See dropouts, silent audio, clipping and stuck PTT remotely
Version, roll out and roll back configuration safely

Four planes, clear responsibilities

A small computer at every site, strict rules

How one transmission travels between sites

Audio transport tuned for speech

Hardware per site

Reliability is part of the design

New capabilities without touching the core

Lessons from the field

The field test surfaced details the lab simulation could not. A talk session that stayed open long after the radio fell silent was traced with a field recording: a small DC offset at the sound card input made silence look like audio and kept VOX open. The agent now filters DC on the capture path; on the same recording VOX closes in 1.8 seconds. Busy repeater traffic also reshaped the alarm logic: back-to-back short transmissions could falsely trigger the "stuck transmitter" alarm. The alarm now uses the PTT time measured by the agent and only fires after more than 70 seconds of continuous transmit.

Timeline

2026-10 — Scope and architecture decisions
Project specification, acceptance tests (T01–T16) and the first architecture decision records.

2026-10 — Panel, relay and field agent
Control panel, Go media relay and Pi agent; end-to-end lab simulation with two virtual gateways.

2026-10 — Live view and hardware guide
Real-time status and talk history; a step-by-step hardware wiring guide inside the panel.

2026-10 — Extension nodes
DMR bridge, voice assistant and APRS iGate nodes.

2026-10 — Field test
Tried with real radios in two separate analog FM cells. Field recordings directly shaped the agent's audio and alarm settings.

Planning a system that runs in the field?