Novu Project Structure

The open-source notification infrastructure. A sophisticated TypeScript monorepo designed for high throughput, separating API services from asynchronous delivery workers.

Updated 2025-12-30

#novu #typescript #nestjs #react #monorepo #microservices #notifications

PNG PDF Copy Prompt

Project Directory

```
novu/
  apps/  Deployable Serv…
    api/  NestJS Main API
    worker/  Delivery proces…
    ws/  WebSocket serve…
    web/  Dashboard UI (R…
  libs/  Shared Internal…
    dal/  Data Access Lay…
    application-generic/
    design-system/
  packages/  SDKs & Providers
    providers/  50+ integrations
    react/  Frontend SDK
  nx.json
  package.json
```

Repository Info

Repository - novuhq/novu

Stars - 35k+

License - MIT

Last Analyzed - December 2025

Tech Stack

Language - TypeScript

Backend - Node.js (NestJS)

Frontend - React

Database - MongoDB / PostgreSQL

Queue - Redis (BullMQ)

Monorepo Tool - Nx + pnpm

Architecture Notes

Novu uses a **Microservices Monorepo** architecture. The system is split into specialized applications: the `api` handles requests, the `worker` processes notification logic and delivery, and the `ws` handles real-time websockets. They use a **Data Access Layer (DAL)** library to abstract the database, allowing the core business logic to remains agnostic of the underlying storage engine. Asynchronous tasks are managed via a robust Redis-based queue system.

Key Directories

apps/worker/ - The heavy-lifter. This service consumes tasks from the queue, evaluates notification templates, and communicates with external providers.

libs/dal/ - Contains the repository patterns and database schemas. It ensures that data access is consistent across the API and Worker services.

packages/providers/ - The library of integrations. Each provider (SendGrid, Twilio, Slack) is an isolated implementation of a common interface.

Why This Structure?

Novu is a prime example of an event-driven system built with TypeScript. It demonstrates how to scale an application by decoupling ingestion from processing and delivery.