



# Signal Server Project Structure

The server-side backend for the Signal messaging app. A focused, high-performance Java application using the Dropwizard framework and Protocol Buffers.

Updated 2025-12-30

#signal #java #dropwizard #protobuf #messaging #security

 PNG

 PDF

 Copy

 Prompt

## Project Directory

### Signal-Server/

- ▼

📁

**service/**

Main Applicatio...
- 📁

**src/main/java/**

Core Java source
- 📁

**config/**

YAML configurat...
- 📄

pom.xml
- 📁

**websocket-resources/**

Real-time Messa...
- 📁

**integration-tests/**

End-to-end serv...
- 📁

**api-doc/**

API documentati...
- 📄

pom.xml

Root Maven Conf...
- 📄

mvnw

Maven wrapper

## Repository Info

**Repository** - [signalapp/Signal-Server](#)

**Stars** - 15k+

**License** - AGPL-3.0

**Last Analyzed** - December 2025

## Tech Stack

**Language** - Java 21

**Framework** - Dropwizard (Jetty/Jersey/Jackson)

**Serialization** - Protocol Buffers

**Database** - PostgreSQL / FoundationDB

**Build Tool** - Maven

## Architecture Notes

Signal Server is built using the **Dropwizard** framework, which prioritizes stable, battle-tested libraries over 'magic'. The architecture is highly optimized for message delivery. It uses **Protocol Buffers** for efficient serialization across the wire. A key architectural split is between the standard REST services and the **websocket-resources**, which manage the long-lived connections required for real-time message pushing.

## Key Directories

**service/** - Contains the bulk of the application code, including account management, key fetching, and message routing logic.

**websocket-resources/** - A specialized module for handling WebSocket-based communication, which is the primary channel for delivering messages to online clients.

**service/config/** - Dropwizard uses YAML for configuration. This directory contains samples for the complex setup required to run a Signal instance.

## Why This Structure?

Signal Server is a masterclass in 'Focused Engineering'. It avoids bloat and focuses entirely on secure, reliable message routing. Its structure shows how to build a world-class system using a minimalist but robust Java stack.