Jellyfin Project Structure

The volunteer-built media solution. A powerful, cross-platform .NET application structured as a classic enterprise solution with many specialized projects.

Updated 2025-12-30

#jellyfin #csharp #dotnet #media-server #ffmpeg #api

PNG PDF Copy Prompt

Project Directory

```
jellyfin/
  Jellyfin.Server/  Main App Entry …
    Program.cs
    Startup.cs
  Jellyfin.Api/  REST API Contro…
    Controllers/
  MediaBrowser.Controller/  Core Business L…
    Library/
    Session/
    Streaming/
  MediaBrowser.Model/  Shared Data Mod…
  MediaBrowser.Providers/  Metadata Fetche…
  MediaBrowser.MediaEncoding/  FFmpeg & Transc…
  Jellyfin.sln  Solution file
  global.json  .NET SDK version
```

Repository Info

Repository - jellyfin/jellyfin

Stars - 30k+

License - GPL-2.0

Last Analyzed - December 2025

Tech Stack

Language - C# 13

Framework - .NET 9.0

Database - SQLite (Default) / Postgres

Media Engine - FFmpeg

Architecture - Multi-project Monolith

Architecture Notes

Jellyfin follows a classic .NET solution structure. It's a collection of over 20 projects, each with a specific responsibility. The 'MediaBrowser' naming convention is inherited from its predecessor, Emby. This modularity allows the team to isolate complex logic like transcoding (`MediaEncoding`) from the API layer (`Jellyfin.Api`).

Key Directories

MediaBrowser.Controller/ - The most important directory. It contains the logic for everything from managing the media database to handling user sessions and playback streams.

MediaBrowser.MediaEncoding/ - The interface to FFmpeg. It generates the complex commands required to transcode various video formats on-the-fly for different clients.

MediaBrowser.Model/ - Contains the 'Plain Old CLR Objects' (POCOs) that represent the data structure of the entire application, shared between all projects.

Why This Structure?

Jellyfin is a premier example of a high-performance C# application. It handles complex, multi-threaded tasks like real-time transcoding while maintaining a highly organized and testable codebase.