# Appwrite Project Structure

The open-source Backend-as-a-Service (BaaS) for web and mobile. A high-performance PHP application using the Utopia framework and Swoole for async I/O.

Updated 2025-12-30

#appwrite #php #utopia #swoole #baas #docker #microservices

PNG PDF Copy Prompt

## Project Directory

```
appwrite/
  app/ Application ent…
    controllers/ API Route defin…
      api/ Client API
      shared/
      web/
    config/ Service configu…
      services.php
      roles.php
    init/ Initialization …
    http.php HTTP Server ent…
    worker.php Worker entry po…
    realtime.php WebSocket serve…
  src/ Core business l…
    Appwrite/
      Auth/ Authentication …
      Database/ Database abstra…
      Messaging/ Email/SMS/Push
      Utopia/ Framework exten…
    Executor/ Function execut…
  bin/ CLI tools & Spe…
    worker-functions Function execut…
    worker-mails Email worker
    migrate Database migrat…
  docs/ Documentation &…
  composer.json PHP Dependencie…
  docker-compose.yml Orchestration c…
  Dockerfile Main server ima…
```

## Repository Info

**Repository** - appwrite/appwrite

**Stars** - 45k+

**License** - BSD-3-Clause

**Last Analyzed** - December 2025

## Tech Stack

**Language** - PHP 8.3

**Framework** - Utopia (Custom)

**Server** - Swoole (Async PHP)

**Database** - MariaDB

**Cache** - Redis

**Architecture** - Micro-Monolith / Services

## Architecture Notes

Appwrite uses a unique architecture. It's a monolith in terms of code (all in one repo), but it runs as multiple microservices containers (`http`, `worker-functions`, `worker-mails`, `realtime`, etc.). The core logic resides in `src/Appwrite` and is shared across all these services. It relies heavily on **Swoole** for asynchronous I/O, making it significantly faster than traditional PHP applications.

## Key Directories

**app/** - Contains the 'Interface' layer: HTTP controllers, CLI commands, and Worker entry points.

**src/Appwrite/** - The 'Domain' layer. Contains the actual business logic for Auth, Database, Storage, etc.

**bin/** - Executables for starting specific workers. This allows scaling individual components (e.g., adding more function workers) independently.

## Why This Structure?

This is the definitive example of modern, high-performance PHP. By using Swoole and a custom framework (Utopia), Appwrite achieves performance comparable to Node.js or Go. Its container-native architecture demonstrates how to build a scalable platform that is easy to self-host.