# **Wellness Companion AI - Task-Mapped Modular Directory Structure**

## **Root Project Structure**

wellness-companion-ai/

├── docker-compose.yml [Tasks 2, 4, 5, 6, 7, 11, 13, 15] [Orchestrates all services]

├── .env.example [Tasks 3, 15] [Environment variables template]

├── .gitignore [Task 3] [Git ignore patterns]

├── README.md [Task 1] [Project documentation]

└── services/ [Task 1] [Dockerized microservices]

## **Layer 1: Desktop Application Service**

services/desktop-app/

├── Dockerfile [Task 1] [Desktop app containerization]

├── requirements.txt [Task 76] [PyQt6, briefcase dependencies]

├── pyproject.toml [Task 95] [Briefcase configuration]

├── src/

│ ├── core/ [Task 76] [Core orchestrator logic]

│ │ ├── orchestrator/ [Task 77] [Request routing & response handling]

│ │ ├── auth\_manager/ [Task 78] [JWT token management]

│ │ └── config/ [Task 76] [Desktop app configuration]

│ ├── ui/ [Task 76] [PyQt6 interface components]

│ │ ├── main\_window/ [Tasks 80, 81, 82, 83] [Primary application window]

│ │ ├── system\_tray/ [Task 86] [Background tray service]

│ │ ├── floating\_overlay/ [Task 87] [Quick query overlay]

│ │ ├── themes/ [Task 85] [QSS styling & theme engine]

│ │ └── dialogs/ [Tasks 79, 84] [File upload, settings dialogs]

│ ├── handlers/ [Task 76] [Desktop-specific handlers]

│ │ ├── file\_handler/ [Task 81] [Local file operations]

│ │ ├── web\_search\_handler/ [Task 77] [Web search integration]

│ │ └── notification\_handler/ [Task 92] [System notifications]

│ └── utils/ [Task 76] [Desktop utilities]

│ ├── keyring\_manager/ [Task 78] [Secure credential storage]

│ ├── health\_checker/ [Task 77] [Service health validation]

│ └── logger/ [Task 96] [Desktop logging]

├── assets/ [Task 76] [UI assets & resources]

│ ├── icons/ [Task 76] [Application icons]

│ ├── themes/ [Task 85] [QSS theme files]

│ └── fonts/ [Task 85] [Custom fonts]

└── dist/ [Task 95] [Briefcase build outputs]

├── dmg/ [Task 95] [macOS distributables]

├── exe/ [Task 95] [Windows distributables]

└── deb/ [Task 95] [Linux distributables]

## **Layer 2: Core Backend Service**

services/core-backend/

├── Dockerfile [Task 9] [FastAPI service container]

├── requirements.txt [Task 36] [FastAPI, Redis, auth dependencies]

├── src/

│ ├── api/ [Task 36] [FastAPI application]

│ │ ├── endpoints/ [Task 36] [API route definitions]

│ │ │ ├── auth/ [Task 70] [Authentication endpoints]

│ │ │ ├── documents/ [Tasks 40, 41, 42, 43] [Document management APIs]

│ │ │ ├── chat/ [Task 89] [Conversation APIs]

│ │ │ ├── search/ [Tasks 39, 44, 45] [Search & knowledge APIs]

│ │ │ └── system/ [Task 38] [Health & admin APIs]

│ │ ├── middleware/ [Task 36] [Custom middleware]

│ │ │ ├── rate\_limiter/ [Task 73] [Redis-based rate limiting]

│ │ │ ├── cors\_handler/ [Task 49] [CORS configuration]

│ │ │ └── error\_handler/ [Task 47] [Global error handling]

│ │ ├── dependencies/ [Task 36] [FastAPI dependencies]

│ │ │ ├── auth\_dependency/ [Task 69] [JWT validation]

│ │ │ └── db\_dependency/ [Task 53] [Database connections]

│ │ └── websocket/ [Task 89] [WebSocket endpoints]

│ ├── services/ [Task 36] [Business logic services]

│ │ ├── auth\_service/ [Tasks 66, 68] [OAuth & JWT management]

│ │ ├── document\_service/ [Task 64] [Document processing coordination]

│ │ ├── chat\_service/ [Task 89] [Conversation management]

│ │ └── search\_service/ [Task 64] [Search orchestration]

│ ├── models/ [Task 46] [Pydantic models]

│ │ ├── auth\_models/ [Task 70] [Authentication schemas]

│ │ ├── document\_models/ [Tasks 40, 41, 42, 43] [Document schemas]

│ │ ├── chat\_models/ [Task 89] [Chat schemas]

│ │ └── response\_models/ [Task 46] [API response schemas]

│ ├── core/ [Task 36] [Core backend utilities]

│ │ ├── config/ [Task 36] [Configuration management]

│ │ ├── security/ [Task 74] [Security utilities]

│ │ └── exceptions/ [Task 47] [Custom exceptions]

│ └── integrations/ [Task 36] [External service integrations]

│ ├── aiml\_client/ [Task 37] [AI/ML service communication]

│ ├── data\_client/ [Task 63] [Data layer communication]

│ └── oauth\_providers/ [Task 67] [Google OAuth, AWS Cognito]

└── tests/ [Task 50] [API tests]

├── unit/ [Task 50] [Unit tests]

└── integration/ [Task 50] [Integration tests]

## **Layer 3: AI/ML Orchestration Service**

services/aiml-orchestration/

├── Dockerfile [Task 8] [AI/ML service container]

├── requirements.txt [Task 16] [LangChain, Ollama, Tavily dependencies]

├── src/

│ ├── orchestrators/ [Task 16] [RAG orchestration logic]

│ │ ├── rag\_orchestrator/ [Task 28] [Main RAG pipeline controller]

│ │ ├── query\_processor/ [Task 26] [Query preprocessing & analysis]

│ │ └── response\_synthesizer/ [Tasks 27, 34] [Response generation & formatting]

│ ├── search/ [Task 16] [Search implementations]

│ │ ├── vector\_search/ [Task 25] [Semantic similarity search]

│ │ ├── hybrid\_search/ [Task 33] [Combined search strategies]

│ │ └── web\_search/ [Task 31] [Tavily API integration]

│ ├── llm/ [Task 16] [Language model integrations]

│ │ ├── ollama\_client/ [Task 24] [Local LLM communication]

│ │ ├── model\_manager/ [Task 24] [Model loading & switching]

│ │ └── prompt\_templates/ [Task 27] [Prompt engineering]

│ ├── embeddings/ [Task 16] [Embedding services]

│ │ ├── sentence\_transformers/ [Tasks 19, 21] [Embedding model integration]

│ │ ├── embedding\_cache/ [Task 62] [Embedding caching layer]

│ │ └── multi\_model\_ensemble/ [Task 19] [Multiple embedding models]

│ ├── reranking/ [Task 16] [Result reranking]

│ │ ├── neural\_rerank/ [Task 29] [Neural reranking models]

│ │ └── confidence\_scorer/ [Task 29] [Confidence scoring algorithms]

│ ├── pipelines/ [Task 16] [Processing pipelines]

│ │ ├── async\_pipeline/ [Task 23] [Asynchronous processing]

│ │ ├── batch\_processor/ [Task 21] [Batch embedding generation]

│ │ └── gpu\_accelerator/ [Task 94] [GPU optimization]

│ └── utils/ [Task 16] [AI/ML utilities]

│ ├── model\_quantization/ [Task 94] [Model optimization]

│ ├── cython\_core/ [Task 94] [Cythonized performance modules]

│ └── metrics/ [Task 98] [Performance monitoring]

└── models/ [Task 7] [Model storage]

├── ollama/ [Task 7] [Local LLM models]

├── embeddings/ [Task 19] [Embedding models]

└── cache/ [Task 62] [Model cache]

## **Layer 4: Data Management Service**

services/data-layer/

├── Dockerfile [Task 10] [Data service container]

├── requirements.txt [Task 51] [Database, vector DB dependencies]

├── src/

│ ├── embedding\_pipeline/ [Task 51] [Document processing pipeline]

│ │ ├── document\_processor/ [Task 17] [PDF, DOCX, TXT processing]

│ │ ├── text\_splitter/ [Task 18] [Chunking strategies]

│ │ ├── embed\_store\_checker/ [Task 59] [Duplicate detection logic]

│ │ └── metadata\_extractor/ [Task 60] [Document metadata extraction]

│ ├── vector\_db/ [Task 51] [Vector database operations]

│ │ ├── qdrant\_client/ [Task 20] [Qdrant operations]

│ │ ├── faiss\_client/ [Task 20] [FAISS operations (alternative)]

│ │ ├── chroma\_client/ [Task 20] [ChromaDB operations (alternative)]

│ │ └── vector\_operations/ [Tasks 22, 61] [CRUD operations]

│ ├── relational\_db/ [Task 51] [PostgreSQL operations]

│ │ ├── models/ [Task 52] [SQLAlchemy models]

│ │ ├── repositories/ [Tasks 54, 55] [Data access layer]

│ │ ├── migrations/ [Task 56] [Database migrations]

│ │ └── connection/ [Task 53] [Database connection management]

│ ├── file\_storage/ [Task 51] [File storage operations]

│ │ ├── s3\_client/ [Task 57] [AWS S3 operations]

│ │ ├── local\_storage/ [Task 57] [Local file operations]

│ │ └── encryption/ [Task 57] [File encryption utilities]

│ ├── cache/ [Task 51] [Caching operations]

│ │ ├── redis\_client/ [Task 62] [Redis operations]

│ │ ├── embedding\_cache/ [Task 62] [Embedding caching]

│ │ └── query\_cache/ [Task 62] [Query result caching]

│ ├── api/ [Task 58] [Data layer API endpoints]

│ │ └── endpoints/ [Tasks 60, 61] [Data processing endpoints]

│ └── sync/ [Task 51] [Data synchronization]

│ ├── cloud\_sync/ [Task 51] [Cloud synchronization]

│ ├── conflict\_resolution/ [Task 51] [Sync conflict handling]

│ └── change\_detector/ [Task 51] [Change tracking]

└── schemas/ [Task 56] [Database schemas]

├── migrations/ [Task 56] [Migration files]

└── seeds/ [Task 56] [Initial data]

## **Layer 5: Infrastructure Service**

services/infrastructure/

├── docker/ [Tasks 4, 5] [Docker configurations]

│ ├── nginx/ [Task 2] [NGINX configurations]

│ │ ├── Dockerfile [Task 2] [NGINX container]

│ │ ├── nginx.conf [Task 2] [Main configuration]

│ │ ├── ssl/ [Task 74] [SSL certificates]

│ │ └── api\_gateway.conf [Task 2] [API gateway rules]

│ ├── redis/ [Task 5] [Redis configurations]

│ │ ├── Dockerfile [Task 5] [Redis container]

│ │ └── redis.conf [Task 5] [Redis configuration]

│ └── postgres/ [Task 4] [PostgreSQL configurations]

│ ├── Dockerfile [Task 4] [PostgreSQL container]

│ ├── init-scripts/ [Task 4] [Initialization scripts]

│ └── backups/ [Task 4] [Backup configurations]

├── aws/ [Task 100] [AWS deployment configurations]

│ ├── terraform/ [Task 100] [Infrastructure as Code]

│ │ ├── ec2/ [Task 100] [EC2 configurations]

│ │ ├── rds/ [Task 100] [RDS configurations]

│ │ ├── elasticache/ [Task 100] [ElastiCache configurations]

│ │ ├── s3/ [Task 100] [S3 bucket configurations]

│ │ └── security\_groups/ [Task 100] [Security group rules]

│ ├── cloudformation/ [Task 100] [CloudFormation templates]

│ └── ecs/ [Task 100] [ECS service definitions]

├── kubernetes/ [Task 100] [K8s configurations (optional)]

│ ├── deployments/ [Task 100] [Service deployments]

│ ├── services/ [Task 100] [Service definitions]

│ ├── ingress/ [Task 100] [Ingress controllers]

│ └── secrets/ [Task 100] [Secret management]

└── monitoring/ [Task 98] [Monitoring configurations]

├── prometheus/ [Task 98] [Prometheus setup]

├── grafana/ [Task 98] [Grafana dashboards]

└── health\_checks/ [Tasks 12, 38] [Health check definitions]

## **Layer 6: CI/CD & Logging Service**

services/cicd-logging/

├── .github/ [Task 99] [GitHub Actions workflows]

│ ├── workflows/ [Task 99] [CI/CD pipelines]

│ │ ├── desktop-build.yml [Task 99] [Desktop app building]

│ │ ├── backend-deploy.yml [Task 99] [Backend deployment]

│ │ ├── docker-publish.yml [Task 99] [Docker image publishing]

│ │ └── security-scan.yml [Task 99] [Security scanning]

│ └── scripts/ [Task 99] [Build scripts]

├── logging/ [Task 96] [Centralized logging]

│ ├── sentry/ [Task 97] [Sentry error tracking]

│ │ ├── config/ [Task 97] [Sentry configurations]

│ │ └── filters/ [Task 97] [Error filtering rules]

│ ├── prometheus/ [Task 98] [Metrics collection]

│ │ ├── exporters/ [Task 98] [Custom exporters]

│ │ └── rules/ [Task 98] [Alerting rules]

│ └── log\_aggregation/ [Task 96] [Log aggregation]

│ ├── fluentd/ [Task 96] [Log forwarding]

│ └── elasticsearch/ [Task 96] [Log storage & search]

├── packaging/ [Task 95] [Application packaging]

│ ├── briefcase/ [Task 95] [Briefcase configurations]

│ ├── pyinstaller/ [Task 95] [PyInstaller specifications]

│ ├── signing/ [Task 95] [Code signing certificates]

│ └── distribution/ [Task 95] [Distribution pipelines]

└── testing/ [Tasks 11, 30, 35, 50, 65, 75, 88, 93] [Testing infrastructure]

├── e2e/ [Task 88] [End-to-end tests]

├── load\_testing/ [Task 98] [Performance tests]

└── security\_testing/ [Tasks 74, 75] [Security tests]

## **Shared Resources**

shared/

├── proto/ [Task 13] [Protocol buffers (if needed)]

├── schemas/ [Task 46] [Shared data schemas]

├── constants/ [Task 36] [Shared constants]

├── utils/ [Task 36] [Shared utilities]

│ ├── crypto/ [Tasks 66, 74] [Encryption utilities]

│ ├── validation/ [Task 46] [Input validation]

│ └── serialization/ [Task 46] [Data serialization]

└── types/ [Task 46] [Type definitions]

## **Task Coverage Summary**

**Phase 1 (Tasks 1-15)**: Foundation & Dockerization

* Infrastructure setup, Docker services, networking, volumes

**Phase 2 (Tasks 16-30)**: Core RAG Foundation

* AI/ML service structure, text processing, embeddings, RAG pipeline

**Phase 3 (Tasks 31-35)**: Web Search Integration

* Tavily API, hybrid search, result synthesis

**Phase 4 (Tasks 36-50)**: Core Backend APIs

* FastAPI setup, all API endpoints, middleware, testing

**Phase 5 (Tasks 51-65)**: Data Persistence

* Data layer service, PostgreSQL, repositories, caching

**Phase 6 (Tasks 66-75)**: Authentication System

* JWT, OAuth, security, multi-user isolation

**Phase 7 (Tasks 76-88)**: Desktop Application

* PyQt6 interface, authentication, all UI components

**Phase 8 (Tasks 89-93)**: Real-time Features

* WebSocket communication, streaming responses

**Phase 9 (Tasks 94-100)**: Production Optimization

* Cython optimization, packaging, logging, monitoring, deployment

Each directory element now shows exactly which tasks are responsible for creating and maintaining that part of the system.