# **Unified Agent Framework Directory Structure**

Here's the project directory structure with a brief description of each active file, its completion status, and what needs to be done:

## **Core Structure**

Copy

unified-agent-framework/

├── src/

│ ├── api/ #

API Layer

│ ├── orchestrator/ #

Task Orchestrator

│ ├── registry/ #

Agent Registry

│ ├── core/ #

Core Agent Infrastructure

│ ├── agents/ #

Specialized Agents

│ ├── db/ #

Database Integration

│ ├── brands/ #

Brand Configuration

│ ├── utils/ #

Utility functions

│ └── index.ts #

Main application entry point

├── scripts/ #

Build and deployment scripts

├── tests/ #

Automated tests

└── config/ #

Configuration files

## **Active Files Status**

### **Core Infrastructure**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/index.ts | Main application entry point | Complete | - |
| src/core/agent.ts | Base agent class | Needs Work | Implement abstract class pattern, add error handling, standardize task result interface |
| src/core/logger.ts | Logging infrastructure | Complete | - |
| src/core/errors.ts | Error handling utilities | Complete | - |
| src/core/metrics.ts | Performance metrics tracking | Needs Work | Implement metrics collection methods |

### **Database Layer**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/db/client.ts | Supabase client initialization | Needs Work | Add connection testing and table initialization |
| src/db/index.ts | Database exports | Complete | - |

### **Agent Registry**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/registry/registry.ts | Agent registration system | Complete | - |
| src/registry/index.ts | Registry exports | Complete | - |

### **Task Orchestrator**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/orchestrator/executor.ts | Task execution manager | Needs Work | Implement task execution flow |
| src/orchestrator/queue.ts | Task queue management | Needs Work | Implement Bull queue integration |
| src/orchestrator/router.ts | Task routing logic | Needs Work | Implement agent selection logic |

### **Brand Configuration**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/brands/index.ts | Brand configuration exports | Complete | - |
| src/brands/config.ts | Brand configuration manager | Needs Work | Implement configuration loading |
| src/brands/schema.ts | Brand schema validation | Needs Work | Define validation schemas |
| src/brands/settings/mysticArcana.ts | Mystic Arcana brand config | Complete | - |
| src/brands/settings/edmShuffle.ts | EDM Shuffle brand config | Complete | - |
| src/brands/settings/birthdayGen.ts | BirthdayGen brand config | Complete | - |

### **Agents**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/agents/index.ts | Agent registration | Complete | - |
| src/agents/file-validator/index.ts | File validation agent | Needs Work | Implement validation methods, add error handling |
| src/agents/duplicate-detector/index.ts | Duplicate file detector | Needs Work | Implement file comparison logic |
| src/agents/tarot-validator/index.ts | Tarot deck validator | Needs Work | Implement deck validation logic |
| src/agents/playlist-validator/index.ts | Playlist validator | Needs Work | Implement playlist validation logic |
| src/agents/message-validator/index.ts | Message template validator | Needs Work | Implement message validation logic |

### **API Layer**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| src/api/index.ts | API route setup | Complete | - |
| src/api/routes/agent.ts | Agent API endpoints | Needs Work | Implement task submission and status endpoints |
| src/api/routes/brand.ts | Brand API endpoints | Needs Work | Implement brand configuration endpoints |
| src/api/middleware/auth.ts | Authentication middleware | Needs Work | Implement JWT validation |
| src/api/middleware/validation.ts | Request validation | Needs Work | Implement request schema validation |

### **Scripts**

| **File** | **Description** | **Status** | **Needs** |
| --- | --- | --- | --- |
| scripts/clean-code.js | Code cleanup utility | Complete | - |
| scripts/optimize-code.js | Code optimization utility | Complete | - |
| scripts/create-test-files.js | Test file generator | Complete | - |
| scripts/create-brand-test-files.js | Brand test file generator | Complete | - |
| scripts/test-api.js | API testing utility | Needs Work | Implement comprehensive API tests |
| scripts/test-brand-agents.js | Brand agent tester | Needs Work | Implement tests for all brand agents |

## **Priority Tasks for Completion**

1. Core Agent Class (  
    src/core/agent.ts  
   )
   * Implement abstract class pattern
   * Add standardized error handling
   * Create TaskResult interface
2. Database Client (src/db/client.ts)
   * Add connection testing
   * Implement table initialization for development
3. Task Orchestrator
   * Complete executor, queue, and router implementations
   * Implement task prioritization
4. Agent Implementations
   * Complete file-validator with validation methods
   * Implement duplicate-detector file comparison logic
   * Finish brand-specific agent implementations
5. API Endpoints
   * Implement task submission and status endpoints
   * Add brand configuration endpoints
   * Add authentication and validation middleware

## **Completion Criteria**

For each file that needs work, the following should be implemented:

1. Type Safety: Proper TypeScript interfaces and types
2. Error Handling: Try/catch blocks with meaningful error messages
3. Logging: Appropriate logging at different levels
4. Documentation: JSDoc comments for public methods
5. Testing: Unit tests for core functionality

Once these are complete, the framework will be ready for initial testing and deployment.

unified-agent-framework

registry.ts