# Enhanced Telegram Trading Bot - Directory Structure

## 📁 Project Root

```

trading-bot/

├── main.py # 🚀 Main application entry point

├── requirements.txt # 📦 Python dependencies (not uploaded)

├── .env # 🔒 Environment variables (not uploaded)

├── README.md # 📖 Project documentation

└── trading\_bot.log # 📝 Application logs (generated)

```

## 📁 Core Packages

### 🔧 API Clients (`clients/`)

```

clients/

├── \_\_init\_\_.py # Package initialization & exports

├── bybit\_client.py # Bybit API client initialization

├── bybit\_helpers.py # Async wrappers for Bybit operations

└── ai\_client.py # OpenAI client initialization

```

### ⚙️ Configuration (`config/`)

```

config/

├── \_\_init\_\_.py # Configuration package exports

├── settings.py # Environment settings & validation

└── constants.py # Trading constants & emoji mappings

```

### 🎮 User Interface (`ui/`)

```

ui/

├── mobile\_layouts.py # Mobile-optimized keyboard layouts

└── keyboards.py # Enhanced keyboard builders

```

### 📊 Dashboard (`dashboard/`)

```

dashboard/

├── \_\_init\_\_.py # Dashboard package exports

└── generator.py # Mobile-optimized dashboard generation

```

### 🎯 Request Handlers (`handlers/`)

```

handlers/

├── \_\_init\_\_.py # Handler registration & setup

├── conversation.py # Enhanced trade setup conversation flow

├── callbacks.py # Callback query handlers

├── commands.py # Command handlers (dashboard, help, etc.)

├── mobile\_handlers.py # Mobile-specific handlers

├── monitoring.py # Position monitoring handlers

└── ai\_handlers.py # AI recommendation handlers

```

### 🚀 Trade Execution (`execution/`)

```

execution/

├── \_\_init\_\_.py # Execution package exports

├── trader.py # Enhanced trade execution logic

├── monitor.py # Position monitoring with performance tracking

└── portfolio\_ai.py # AI-powered portfolio analysis

```

### 🛡️ Risk Management (`risk/`)

```

risk/

├── \_\_init\_\_.py # Risk management exports

├── calculations.py # Risk/reward calculations & position sizing

├── assessment.py # AI-powered risk assessment

├── sentiment.py # Market sentiment analysis

└── regime.py # Market regime detection

```

### 🔧 Utilities (`utils/`)

```

utils/

├── \_\_init\_\_.py # Utilities package exports

├── formatters.py # Mobile-optimized number/text formatting

├── cache.py # Enhanced caching system with TTL

├── helpers.py # General helper functions & decimal handling

├── bybit\_helpers.py # Bybit API helper functions

└── position\_modes.py # Position mode management (hedge/one-way)

```

### 🔄 Shared Components (`shared/`)

```

shared/

├── \_\_init\_\_.py # Shared utilities exports

└── state.py # Message management & state handling

```

## 📁 Data & Configuration Files (Not Uploaded)

```

data/

├── bybit\_bot\_dashboard\_v4.1\_enhanced.pkl # Bot persistence data

└── logs/

└── trading\_bot.log # Application logs

```

## 📁 External Dependencies

```

requirements.txt # Should include:

├── python-telegram-bot>=20.0

├── pybit>=5.0

├── openai>=1.0

├── python-dotenv

├── aiohttp

└── asyncio

```

## 📁 Environment Configuration

```

.env # Should include:

├── TELEGRAM\_TOKEN=your\_bot\_token

├── BYBIT\_API\_KEY=your\_api\_key

├── BYBIT\_API\_SECRET=your\_api\_secret

├── USE\_TESTNET=false

├── OPENAI\_API\_KEY=your\_openai\_key

└── LLM\_PROVIDER=openai

```

## 🎯 Key Features by Package

### 📱 \*\*Mobile-First Design\*\*

- `ui/mobile\_layouts.py` - iPhone 16 Pro Max optimized layouts

- `ui/keyboards.py` - Touch-friendly button arrangements

- `utils/formatters.py` - Mobile-optimized text formatting

### 🚀 \*\*Enhanced Trading\*\*

- `execution/trader.py` - Dual approach trading (Fast Market vs Conservative Limits)

- `handlers/conversation.py` - Step-by-step guided trade setup

- `risk/calculations.py` - Smart position sizing & R:R analysis

### 🧠 \*\*AI-Powered Insights\*\*

- `risk/assessment.py` - AI risk scoring

- `risk/sentiment.py` - Market sentiment analysis

- `execution/portfolio\_ai.py` - Portfolio health optimization

### 🔄 \*\*Advanced Monitoring\*\*

- `execution/monitor.py` - Real-time position tracking

- `handlers/monitoring.py` - Automated monitoring setup

- `utils/cache.py` - Performance-optimized caching

### 🛡️ \*\*Risk Management\*\*

- `risk/calculations.py` - Risk/reward calculations

- `utils/position\_modes.py` - Hedge mode support

- `clients/bybit\_helpers.py` - Orphaned order cleanup

## 🚀 \*\*Getting Started\*\*

1. \*\*Setup Environment:\*\*

```bash

pip install -r requirements.txt

cp .env.example .env # Configure your API keys

```

2. \*\*Run the Bot:\*\*

```bash

python main.py

```

3. \*\*Available Commands:\*\*

- `/start` - Enhanced dashboard

- `/trade` - Manual trade setup

- `/hedge\_mode` - Enable hedge mode

- `/help` - User guide

## 📋 \*\*Architecture Highlights\*\*

✅ \*\*Dual Trading Approaches:\*\* Fast Market vs Conservative Limits

✅ \*\*Mobile-First UX:\*\* Touch-optimized for iPhone 16 Pro Max

✅ \*\*AI-Enhanced:\*\* Smart risk assessment & market analysis

✅ \*\*Real-time Monitoring:\*\* Automatic position tracking

✅ \*\*Advanced Risk Management:\*\* R:R calculations & position sizing

✅ \*\*Robust Error Handling:\*\* Retry logic & fallback mechanisms

✅ \*\*Performance Optimized:\*\* Caching & efficient data structures

✅ \*\*Extensible Design:\*\* Modular architecture for easy enhancements