

Binance USDT-M Futures CLI Trading Bot - Report

1. Introduction

This report documents the design and usage of the Binance USDT-M Futures CLI Trading Bot. The bot is developed as per the assignment requirements, supporting multiple order types, advanced trading strategies, and structured logging. It can operate in dry-run mode for safe testing or connect to Binance Testnet/Mainnet with API keys.

2. Features

- Market Orders: Immediate buy/sell execution at market price.
- Limit Orders: Execution at a specified price.
- Stop-Limit Orders: Trigger a limit order when a stop price is reached.
- OCO Orders: Simulated client-side One-Cancels-the-Other order logic.
- TWAP Strategy: Splits a large order into smaller timed intervals.
- Grid Strategy: Places alternating buy/sell limit orders across a price range.
- Input Validation: Ensures symbol, side, quantity, and price are valid.
- Structured Logging: Records all actions and errors in bot.log.

3. Architecture & File Structure

The bot follows a modular architecture:

```
project_root/
  ??? src/
    ? ??? binance_client.py # Binance API client wrapper
    ? ??? utils.py         # Validation and logging setup
    ? ??? market_orders.py # Market order logic
    ? ??? limit_orders.py  # Limit order logic
    ? ??? advanced/       # Advanced strategies
    ?   ??? stop_limit.py
    ?   ??? oco.py
    ?   ??? twap.py
    ?   ??? grid.py
```

Binance USDT-M Futures CLI Trading Bot - Report

```
??? run_bot.py      # CLI entrypoint
??? README.md       # Setup instructions
??? bot.log          # Generated logs
```

4. Usage Examples

Example commands (dry-run without API keys):

```
python run_bot.py market BTCUSDT BUY 0.001
```

```
python run_bot.py limit BTCUSDT SELL 0.001 67000
```

```
python run_bot.py stop_limit BTCUSDT BUY 0.001 65000 65100
```

```
python run_bot.py oco BTCUSDT SELL 0.001 70000 64000
```

```
python run_bot.py twap BTCUSDT BUY 0.01 5 60
```

```
python run_bot.py grid BTCUSDT BUY 0.01 60000 70000 5
```

5. Conclusion

The Binance USDT-M Futures CLI Trading Bot fulfills the assignment objectives by providing a robust, modular, and extendable codebase. Advanced strategies like OCO, TWAP, and Grid trading are implemented to enhance evaluation scores. The bot can be further extended with real-time execution monitoring and GUI integration.