**📌 Чек-лист боеготовности AI-Трейдера (заполнен по коду репозитория)**

Ниже — ваш чек-лист, заполненный после просмотра каждого файла из архива репо. Для каждого пункта даю статус (✅/⚠️/❌), приоритет (P0/P1/P2) и **доказательство** с указанием путей/функций.

**1) Базовые цели и принципы**

* ✅ **Автономная работа 24/7** (P1)  
  Доказательство: Docker/compose с restart: always, healthcheck, supervisord для без-докерных деплоев.  
  Файлы: Dockerfile, docker-compose.yml, deploy/supervisord.conf.
* ✅ **Приоритет — риск и стабильность** (P0)  
  Доказательство: централизованный риск-конфиг с клампингом параметров, дневные лимиты, dead-man.  
  Файлы: utils/risk\_config.py (RiskConfig), state/daily\_limits.py, risk/deadman.py.
* ✅ **Полный аудит действий** (P1)  
  Доказательство: журнал ордеров и CRUD API для него.  
  Файлы: db/models\_orders.py, db/crud\_orders.py, логирование по модулям.
* ✅ **Режимы demo/real** (P0)  
  Доказательство: исполняющий класс с testnet=True/False, выбор режима через UI/ENV.  
  Файлы: executors/api\_binance.py (BINANCE\_DOMAIN\_TEST/MAIN), routers/ui.py (параметры mode, testnet), services/trading\_service.py (mode="sim"|"binance", testnet).

**2) Инфраструктура**

* ✅ **Docker/Compose** (P1)  
  Файлы: Dockerfile, docker-compose.yml.
* ⚠️ **Автоперезапуски (supervisord/systemd)** (P1)  
  Есть deploy/supervisord.conf. **systemd unit** отсутствует (можно добавить deploy/systemd/ai-trader.service).
* ✅ **Бэкапы** (P2)  
  Файлы: backup/backup.sh, backup/crontab.txt. (Расписания/алертов об ошибках нет — добавить.)
* ❌ **Мониторинг ресурсов (CPU/RAM/NET)** (P2)  
  Нет экспортера/прометей-метрик. (Добавить psutil/Prometheus client.)
* ✅ **Автовосстановление/State-recovery** (P1)  
  Файлы: services/reconcile.py (сверка, чистка «висячих»), sources/binance\_ws.py (backoff/реконнект).

**3) Источники данных**

* ✅ **Binance WebSocket/REST** (P0)  
  Файлы: sources/binance.py (REST OHLCV + пагинация/ретраи), sources/binance\_ws.py (WS с heartbeat/backoff).
* ✅ **OHLCV сбор/хранение** (P1)  
  Файлы: sources/ohlcv.py, db/models.py, миграции: migrations/.
* ⚠️ **Новости RSS/TG/Twitter** (P2)  
  Есть каркас RSS/NLP, **не подключено** в боевую петлю сигналов.  
  Файлы: news/rss\_client.py, news/nlp\_gate.py.
* ❌ **Синхронизация времени (UTC+NTP)** (P2)  
  Проверок дрейфа нет.
* ✅ **Backfill при обрывах** (P1)  
  Файлы: sources/binance.py (fetch\_klines с пагинацией по openTime и ретраями).

**4) Анализ и Feature Engineering**

* ✅ **Индикаторы (EMA/RSI/MACD/ATR/Bollinger)** (P1)  
  Файлы: src/indicators.py, тесты: tests/test\_indicators.py.
* ⚠️ **Свечные модели (Hammer/Doji/Engulfing)** (P2)  
  Явного модуля нет — использовать в качестве фич в src/analysis/analyze\_market.py отсутствует. (Добавить src/features/patterns.py.)
* ⚠️ **Сентимент (BERT/BiLSTM)** (P2)  
  Модели подключения нет; каркас источников есть (news/\*).
* ❌ **Fractional diff / Triple-barrier** (P2)  
  Нет реализаций/тестов.
* ⚠️ **Мультифреймовый анализ (Элдер)** (P1)  
  Частично: в анализе есть режимы рынка/волатильность/тренд-gap, можно расширить MTF входы.  
  Файлы: src/analysis/analyze\_market.py, src/strategy.py (режимы: is\_trend/is\_flat/is\_turbulent).

**5) Модели и прогнозы**

* ❌ **LSTM/Transformer (price/vol)** (P0)  
  Нет боевого инференса/весов (models/ отсутствует).
* ❌ **RF/XGBoost (классификация сигналов)** (P1)  
  Нет интеграции.
* ❌ **DRL-агент** (P2)  
  Нет.
* ❌ **Autoencoder/GAN (факторы/стресс)** (P2)  
  Нет.
* ❌ **Meta-labeling** (P2)  
  Нет.

Примечание: базовый ско́ринг сигналов реализован без ML:  
src/strategy.py (веса признаков, make\_signal\_matrix/aggregate\_signals).

**6) Генерация торговых сигналов**

* ⚠️ **Ансамбль стратегий (trend/mean-reversion/event-driven)** (P1)  
  Есть базовая стратегия и агрегатор/режимы; ансамбля/конфига весов нет.  
  Файлы: src/strategy.py, src/analysis/analyze\_market.py.
* ⚠️ **Confidence score** (P1)  
  Есть score от взвешенной суммы признаков (src/strategy.py), но нет единых порогов/калибровки через валидацию.
* ✅ **Фильтр режимов рынка** (P1)  
  Файлы: src/strategy.py (is\_trend/is\_flat/is\_turbulent).
* ⚠️ **Контроль частоты сделок** (P1)  
  Ограничения на день в рисках есть (max\_trades\_per\_day), но **частота/тайм-гейт** на уровне стратегии не задан.  
  Файлы: utils/risk\_config.py → max\_trades\_per\_day, учёт в services/trading\_service.py.

**7) Управление рисками**

* ✅ **≤2% на сделку** (P0)  
  Доказательство: конфиг по умолчанию 1% (risk\_pct\_per\_trade=0.01), расчёт объёма по стоп-дистанции.  
  Файлы: utils/risk\_config.py, services/trading\_service.py (\_check\_risk\_and\_size).
* ✅ **≤6% суммарно** (P0)  
  Доказательство: portfolio\_max\_risk\_pct=0.06, суммирование риска по открытым позициям.  
  Файлы: utils/risk\_config.py, services/trading\_service.py (portfolio\_risk\_used, can\_open\_new).
* ⚠️ **Авто-размер позиции (ATR/Optimal-f)** (P1)  
  Есть расчёт из стоп-дистанции и risk\_pct\_per\_trade. **Optimal-f** отсутствует.  
  Файлы: services/trading\_service.py.
* ✅ **SL/TP и трейлинг при входе** (P0)  
  Доказательство: единый вызов open\_with\_protection (OCO/SL/TP), трейлинг настроен в рисках.  
  Файлы: executors/api\_binance.py (open\_with\_protection → OCO/SL/TP), risk/trailing.py.
* ✅ **Dead-man switch** (P1)  
  Файлы: risk/deadman.py (heartbeat, deadman\_max\_stale\_sec).
* ✅ **Дневные убытки (стоп-день)** (P0)  
  Файлы: state/daily\_limits.py, utils/risk\_config.py (daily\_max\_loss\_pct).

**8) Исполнение сделок**

* ✅ **Binance API** (P0)  
  Файлы: executors/api\_binance.py (REST с ретраями/лимитами/фильтрами символов).
* ✅ **Рыночные/лимитные/OCO** (P0)  
  Файлы: executors/api\_binance.py (post\_order, post\_oco\_order внутри open\_with\_protection).
* ⚠️ **Учёт комиссий/проскальзывания** (P1)  
  В симуляторе — да (src/paper.py: fee\_pct), в реальной сделке — комиссия implicit через биржу, но **PnL пересчёты** в UI/БД стоит явнее учитывать.  
  Файлы: executors/api\_binance.py, db/crud\_orders.py, templates/monitor/\_metrics.html.
* ✅ **Автосверка позиций** (P1)  
  Файлы: services/reconcile.py (нетто-позиции/подтягивание ордеров).
* ✅ **Закрытие/реверс** (P1)  
  Файлы: executors/api\_binance.py (close\_order, close\_all), routers/trading\_exec.py (ендпойнты close\_order, cancel\_order).

**9) Сопровождение сделок**

* ✅ **Мониторинг позиций (RT)** (P0)  
  Доказательство: UI partials дергают /exec/list\_positions и /exec/balance.  
  Файлы: routers/trading\_exec.py (list\_positions, get\_balance), routers/ui.py (/ui/partials/...), templates/monitor/\_positions.html, \_balance.html.
* ✅ **Автоподтягивание стопов** (P1)  
  Файлы: risk/trailing.py.
* ⚠️ **Частичное закрытие / мульти-активы** (P1)  
  Частичное — поддержано параметром qty в close\_order; мульти-актив — да; **нет сценарного теста**.  
  Файлы: executors/api\_binance.py (close\_order), services/reconcile.py.
* ⚠️ **Учёт комиссий в PnL** (P1)  
  В paper — есть; в прод-цепочке PnL/метрики нужны явные формулы с комиссиями.  
  Файлы: src/paper.py, templates/monitor/\_metrics.html.

**10) Мониторинг и отчёты**

* ✅ **Веб-дашборд** (P1)  
  Файлы: templates/monitor/\*, routers/ui.py (HTMX partials).
* ⚠️ **Метрики: PnL/Equity/Sharpe/Drawdown** (P1)  
  Есть расчёт метрик на основе OHLCV выборки (UI), но нет полной онлайновой связки с реальными сделками и комиссиями.  
  Файлы: routers/ui.py (\_compute\_metrics\_close), templates/monitor/\_metrics.html.
* ⚠️ **Алерты (TG/E-mail)** (P2)  
  Каркас есть, но интеграция с триггерами/ENV не показана.  
  Файлы: monitoring/alerts.py.
* ❌ **Автоотчёты (день/неделя)** (P2)  
  Нет крон-тасков/генерации отчётов.
* ✅ **Журнал сделок** (P1)  
  Файлы: db/crud\_orders.py, db/models\_orders.py (+ UI вывод последних ордеров routers/ui.py → /partials/orders).

**11) Тестирование и контроль качества**

* ✅ **Юнит-тесты (OHLCV/стратегия/исполнение/UI)** (P1)  
  Файлы: tests/test\_ohlcv.py, tests/test\_analyze\_market.py, tests/test\_strategy.py, tests/test\_execution\_api.py, tests/test\_ui\_smoke.py, др.
* ✅ **Backtest с комиссиями/проскальзыванием** (P1)  
  Файлы: src/paper.py (+ tests/test\_paper.py).
* ⚠️ **Paper trading на Testnet** (P1)  
  Реальные эндпойнты есть; отдельного длительного e2e-теста на testnet **нет**. (Добавить tests/test\_e2e\_testnet.py.)
* ❌ **Purged Walk-Forward CV для моделей** (P2)  
  Нет ML-пайплайнов.
* ❌ **Стресс-тесты (GAN-рынки)** (P2)  
  Нет.

**12) Безопасность**

* ⚠️ **Хранение ключей (.env) / шифрование** (P0)  
  Загрузка из .env/YAML (utils/secrets.py), **без шифрования**. (Рекомендую OS keyring / KMS.)  
  Файлы: configs/.env, utils/secrets.py.
* ⚠️ **Маскирование секретов в UI** (P1)  
  Не обнаружено маскирующих фильтров на шаблонах (ключи не показываются, но стоит гарантировать).
* ❌ **Двухуровневое подтверждение критических действий** (P1)  
  Нет PIN/OTP на «закрыть всё»/«cancel all».
* ✅ **Логи и аудит** (P1)  
  Широкое логирование по сервисам, журнал ордеров в БД.  
  Файлы: db/crud\_orders.py, логи в модулях executors/\*, services/\*.

**Резюме статусов по чек-листу**

| **Раздел** | **Готово** | **Частично** | **Нет** |
| --- | --- | --- | --- |
| 1. Базовые принципы | ✅ |  |  |
| 2. Инфраструктура | ✅ | ⚠️ | ❌ |
| 3. Источники данных | ✅ | ⚠️ |  |
| 4. Анализ/Фичи | ✅ | ⚠️ | ❌ |
| 5. Модели/ML |  |  | ❌ |
| 6. Сигналы |  | ✅ |  |
| 7. Риски | ✅ | ⚠️ |  |
| 8. Исполнение | ✅ | ⚠️ |  |
| 9. Сопровождение | ✅ | ⚠️ |  |
| 10. Мониторинг/отчёты | ✅ | ⚠️ | ❌ |
| 11. Тестирование | ✅ | ⚠️ | ❌ |
| 12. Безопасность | ✅ | ⚠️ | ❌ |

**P0 — что блокирует «полный автотрейдинг-цикл» прямо сейчас**

1. **ML-моделей нет** — цикл работает на правилах/индикаторах. (Можно оставить как есть для автотрейдинга; ML — P1/P2.)
2. **Явный учёт комиссий в PnL/UI** — привести к единой формуле (как в paper.py).
3. **Risk-gate железобетонно перед исполнением** — уже есть, но зафиксировать обязательный вызов в сервисе перед любым open\_\*.  
   Файлы: services/trading\_service.py (\_check\_risk\_and\_size, open\_market/open\_limit).
4. **E2E testnet тест** «анализ → open → SL/TP → close → журнал/баланс» — добавить tests/test\_e2e\_testnet.py.
5. **Безопасность** — увести ключи из простого .env в защищённое хранилище; добавить 2FA на критические действия.