Инструкция по установке и развертыванию

1. Поддерживаемая конфигурация

Фактически проверенная серверная конфигурация:

- архитектура x86-64;
- Ubuntu 22.04.5 LTS;
- ядро Linux 5.15.0-43-generic для проверенного стека AC700;
- CPython 3.10.12;
- FastAPI 0.159.1, Uvicorn 0.54.0, HTTPX 0.28.1;
- Docker 29.1.3 для управляемого контейнерного режима;
- STC/Teco AC700 и tecovLLM 3.2.0 для проверенного аппаратного профиля.

Прямой режим веб-приложения не требует Docker или AC700. Для него требуется доступный сервер модели, реализующий нужную часть OpenAI API.

2. Предварительные условия

Системный администратор должен подготовить:

- Linux-сервер с Python 3.10;
- отдельную системную учетную запись приложения;
- исходный код или дистрибутив ПО;
- Python-пакеты из requirements.txt;
- OpenAI-совместимый сервер модели;
- при публикации в сети: обратный прокси, DNS и TLS-сертификат;
- при использовании шлюза: Docker, профиль модели и среду вывода.

Для установки в изолированном контуре зависимости и контейнерные образы следует заранее перенести во внутренний репозиторий или на носитель.

3. Установка веб-приложения

Пример каталога установки: /opt/local-llm. Конкретный каталог может быть выбран администратором.

install -d -o root -g root -m 0755 /opt/local-llm

Разместить в /opt/local-llm содержимое утвержденного дистрибутива.

cd /opt/local-llm

python3 -m venv venv

venv/bin/pip install --no-cache-dir -r requirements.txt

cp .env.example .env

chmod 0640 .env

Проверить контрольную сумму и версию дистрибутива до установки. Не следует выполнять git pull непосредственно в продуктивной среде без процедуры выпуска и резервной копии.

4. Настройка прямого режима

Минимальный пример .env:

APP_NAME="ПК «ЦСТ.Локальный LLM-шлюз»"

LLM_BASE_URL=http://127.0.0.1:8000/v1

LLM_API_KEY=

GATEWAY_ADMIN_ENABLED=false

DB_PATH=/opt/local-llm/data/chat.db

HOST=127.0.0.1

PORT=8130

SESSION_SECRET=<сгенерированная случайная строка>

MODELS=

REQUEST_TIMEOUT=900

COOKIE_SECURE=true

SESSION_TTL_HOURS=168

MAX_MESSAGE_CHARS=32000

HISTORY_MESSAGES=40

Секрет сессии можно сформировать локально:

python3 -c 'import secrets; print(secrets.token_hex(32))'

Если сервер модели требует Bearer-токен, он задается в LLM_API_KEY. Значение нельзя публиковать в документации или Git.

5. Создание учетной записи администратора

```
cd /opt/local-llm

set -a

. ./.env

set +a

venv/bin/python -m app.cli create-user admin --admin
```

Программа запросит пароль дважды. Минимальная длина пароля составляет восемь символов.

6. Пробный запуск

```
cd /opt/local-llm

set -a

. ./.env

set +a
```

venv/bin/uvicorn app.main:app --host 127.0.0.1 --port 8130

Проверка:

curl -fsS http://127.0.0.1:8130/api/config

curl -fsS http://127.0.0.1:8130/healthz

В /healthz проверка upstream.models.ok должна иметь значение true, если сервер модели доступен.

7. Системная служба

Создать системного пользователя без интерактивного входа, предоставить ему чтение каталога приложения и запись только в каталог данных. Адаптировать пути в deploy/own-chat.service, затем выполнить:

cp deploy/own-chat.service /etc/systemd/system/own-chat.service

systemctl daemon-reload

systemctl enable --now own-chat.service

systemctl status own-chat.service

Файл примера включает ограничения systemd: NoNewPrivileges, ProtectSystem, ProtectHome, запрет изменения модулей ядра и разрешение записи только в каталог данных.

8. HTTPS

Пример конфигурации обратного прокси находится в deploy/Caddyfile.example. Перед применением заменить доменное имя и адреса на используемые в вашей инфраструктуре. Публично следует закрыть внутренние пути /healthz, /metrics и /v1/*.

Использование Caddy не обязательно. Допускается другой обратный прокси с поддержкой HTTPS и потоковой передачи без буферизации.

9. Установка собственного шлюза

Шлюз нужен только при управлении несколькими серверами моделей или их запуском из профилей.

- Подготовить сервер вывода и проверить его отдельно.
- Создать JSON-профиль модели на основе структуры gateway/models.json.
- Выбрать TELEMETRY_PROVIDER: поддержанный адаптер или none.
- Создать случайный Bearer-ключ в API_KEYS.
- Установить файл окружения и unit-файл.

cp deploy/ac700-gateway.env.example /etc/model-gateway.env

chmod 0600 /etc/model-gateway.env

Адаптировать EnvironmentFile и описание в unit-файле.

cp deploy/ac700-gateway.service /etc/systemd/system/model-gateway.service

systemctl daemon-reload

systemctl enable --now model-gateway.service

В .env веб-приложения затем указать:

LLM_BASE_URL=http://127.0.0.1:8120/v1

LLM_API_KEY=<Bearer-ключ, созданный при настройке шлюза>

GATEWAY_ADMIN_ENABLED=true

GATEWAY_ROOT_URL=http://127.0.0.1:8120

10. Проверенный профиль AC700

Для профиля из gateway/models.json дополнительно требуются:

- четыре устройства /dev/tcaicard0 ... /dev/tcaicard3;

- работоспособная команда /opt/tecoai/bin/teco-smi -c;

- контейнерный образ tecovLLM 3.2.0 (получить из локального реестра или с носителя);

- модели в путях, указанных в профиле;

- Docker-сеть, указанная в профиле;

- TELEMETRY_PROVIDER=teco-smi.

Перед запуском проверить:

ls -l /dev/tcaicard*

/opt/tecoai/bin/teco-smi -c

Проверить наличие контейнерного образа tecovLLM 3.2.0 в локальном реестре или на носителе. Имя образа и способ его получения определяются внутренней инфраструктурой организации.

Версия ядра 5.15.0-43-generic относится к текущей проверенной установке драйвера AC700. Она не является общим требованием веб-приложения.

11. Приемочная проверка

После установки выполнить:

python3 -m compileall -q app gateway tools

python3 -m unittest discover -v

curl -fsS http://127.0.0.1:8130/api/config

curl -fsS http://127.0.0.1:8130/healthz

Затем войти через браузер, создать чат, отправить тестовый запрос, дождаться потокового ответа, обновить страницу и убедиться, что история сохранена.

12. Обновление и откат

Перед обновлением:

- зафиксировать текущую версию;
- создать и проверить резервную копию SQLite;
- сохранить файлы окружения и unit-файлы;
- установить утвержденный выпуск;
- выполнить тесты и перезапустить только измененные службы;

- проверить вход, историю и генерацию.

При неудаче вернуть предыдущий дистрибутив и базу. Подробная процедура отката описана в руководстве администратора (раздел 6).