Table of Contents

Отлично 💼 Вот готовый документ в формате Markdown — **Legal Assistant Arbitrage DevOps Guide v2.4** — единый PDF-ориентированный гайд, собранный из всех предыдущих DevOps-документов. Ты можешь конвертировать его в PDF командой:

pandoc "Legal Assistant Arbitrage DevOps Guide v2.4.md" -o "Legal Assistant Arbitrage DevOps Guide v2.4.pdf" --standalone --toc

# ⚖️ Legal Assistant Arbitrage — DevOps Guide v2.4  
  
---  
  
## 🏛️ Общая информация  
  
\*\*Проект:\*\* Legal Assistant Arbitrage   
\*\*Версия:\*\* v2.4   
\*\*Дата выпуска:\*\* 2025-10-09   
\*\*Авторы:\*\*   
- DevOps Lead — `admin@legal-assistant`   
- Infrastructure Owner — `alex@legal-assistant.tech`   
- CI/CD Maintainer — `ci-bot@github`  
  
\*\*Назначение:\*\*   
Данное руководство описывает архитектуру, пайплайн, тестовую инфраструктуру,   
операции, ручной деплой, откат и логи релизов проекта \*\*Legal Assistant Arbitrage v2.4\*\*.   
Это корпоративный документ, объединяющий все DevOps-процессы и протоколы работы.  
  
---  
  
## 📚 Оглавление  
  
1. CI/CD Pipeline Overview   
2. Test Guide   
3. DevOps Operations Manual   
4. Deploy Manual   
5. Rollback Manual   
6. Deploy Log Template   
7. DevOps Principles   
8. Контакты и ответственность  
  
---  
  
## 🚀 CI/CD Pipeline Overview  
  
\*\*Расположение:\*\* `.github/workflows/ci.yml`   
\*\*Описание:\*\* Автоматизированный CI/CD пайплайн, обеспечивающий полный цикл от тестов до деплоя.  
  
### Этапы пайплайна  
1. \*\*lint-and-test\*\* — линтинг, юнит-тесты, миграции   
2. \*\*smoke\*\* — проверка `/api/health`   
3. \*\*integration\*\* — проверка Telegram и KAD   
4. \*\*build-and-push\*\* — сборка Docker-образа и публикация в GHCR   
5. \*\*deploy\*\* — развертывание на staging/production   
  
### Telegram-уведомления  
Все этапы CI/CD сопровождаются уведомлениями:  
- ✅ успешные тесты   
- ❌ ошибки   
- 🧩 интеграционные проверки   
- 🚀 деплой завершён   
  
\*\*Secrets, используемые в CI:\*\*

TELEGRAM\_BOT\_TOKEN TELEGRAM\_CHAT\_ID PROD\_SERVER\_HOST / STAGING\_SERVER\_HOST PROD\_SERVER\_SSH\_KEY / STAGING\_SERVER\_SSH\_KEY

---  
  
## 🧪 Test Guide  
  
\*\*Расположение:\*\* `docs/TEST\_GUIDE.md`   
\*\*Назначение:\*\* Описывает классификацию и запуск тестов.  
  
| Тип | Команда | Назначение |  
|------|----------|------------|  
| Unit | `pytest -v` | Проверка бизнес-логики |  
| Smoke | `pytest -m smoke` / `make smoke` | Проверка `/api/health` |  
| Integration | `pytest -m integration` / `make integration` | Проверка Telegram и KAD |  
| Manual | `make telegram-notify-test` | Ручная проверка уведомлений |  
  
\*\*Основные Makefile команды:\*\*

make smoke make smoke-local make integration make integration-local

\*\*Рекомендации:\*\*  
- Использовать `--maxfail=1` и `--tb=short` для CI   
- Перед пушем в `main` убедиться, что smoke и integration проходят зелёно   
- Все integration тесты должны иметь пометку `@pytest.mark.integration`  
  
---  
  
## 🛠️ DevOps Operations Manual  
  
\*\*Расположение:\*\* `docs/DEVOPS\_OPERATIONS.md`   
\*\*Назначение:\*\* Подробное руководство по DevOps-процессам.  
  
### Основные команды  
| Команда | Описание |  
|----------|-----------|  
| `make up` | Запуск контейнеров |  
| `make down` | Остановка контейнеров |  
| `make migrate` | Применение миграций |  
| `make doctor` | Проверка окружения |  
| `make backup-db` | Резервное копирование БД |  
| `make integration` | Проверка внешних сервисов |  
  
### База данных  
- Снятие дампа:   
 ```bash  
 docker exec -t legal-assistant-db pg\_dump -U admin legal\_assistant\_db > artifacts/db\_backup.sql

* Восстановление:
* cat artifacts/db\_backup.sql | docker exec -i legal-assistant-db psql -U admin -d legal\_assistant\_db

### Проверка здоровья

curl -s http://127.0.0.1:8080/api/health | jq

## 🚀 Deploy Manual

**Расположение:** docs/DEPLOY\_MANUAL.md **Назначение:** Пошаговый деплой проекта вручную.

### Этапы

1. Подключение по SSH к серверу:

* ssh -i ~/.ssh/prod\_legal\_assistant.pem user@server-ip

1. Обновление кода:

* git pull origin main

1. Пересборка контейнеров:

* docker compose -f docker-compose.prod.yml up -d --build

1. Применение миграций:

* make migrate

1. Проверка API:

* make smoke-local

1. Проверка интеграций:

* make integration-local

1. Уведомление в Telegram:

* python3 backend/app/utils/notify\_telegram.py "🚀 Ручной деплой завершён успешно ✅"

## 🔄 Rollback Manual

**Расположение:** docs/ROLLBACK\_MANUAL.md **Назначение:** Алгоритм отката версии.

### Этапы rollback:

1. Бэкап БД:

* docker exec -t legal-assistant-db pg\_dump -U admin legal\_assistant\_db > artifacts/rollback\_backup.sql

1. Откат Docker-образа:

* docker pull ghcr.io/aw56/legal-assistant-backend:v2.3  
  docker compose -f docker-compose.prod.yml up -d --force-recreate

1. Откат миграций:

* docker compose -f docker-compose.prod.yml exec -T backend alembic downgrade -1

1. Smoke-проверка и уведомление:

* make smoke-local  
  make telegram-notify-test

**Telegram сообщение:**

♻️ Откат выполнен: Legal Assistant Arbitrage v2.4 → v2.3 успешно завершён ✅

## 🧾 Deploy Log Template

**Расположение:** docs/DEPLOY\_LOG\_TEMPLATE.md **Назначение:** Формат журналирования всех деплоев и rollback-ов.

### Пример записи:

🗓️ Дата: 2025-10-09 18:45 UTC  
👤 Исполнитель: GitHub Actions  
🔖 Версия: v2.4.1  
🌿 Ветка: main  
⚙️ Тип: Deploy  
📦 Образ: ghcr.io/aw56/legal-assistant-backend:v2.4.1  
📊 Результат: ✅ Успешно  
Описание: Добавлена интеграция Telegram и KAD

**Файл журнала:** docs/DEPLOY\_LOG.md (дополняется автоматически CI или вручную после ручного деплоя)

## ⚙️ DevOps Principles

**Основные принципы DevOps в проекте:**

| Принцип | Суть |
| --- | --- |
| **Automation First** | Все операции выполняются через Makefile или CI |
| **Fail Fast** | Ошибка теста мгновенно останавливает пайплайн |
| **Immutable Deploys** | Каждый деплой использует уникальный Docker-тэг |
| **Continuous Verification** | Smoke и integration проверяют каждое обновление |
| **ChatOps Integration** | Telegram — основной канал обратной связи DevOps |
| **Transparency** | Все деплои и откаты логируются в DEPLOY\_LOG.md |

## 📞 Контакты и ответственность

| Роль | Ответственный | Контакт |
| --- | --- | --- |
| DevOps Lead | Алексей | admin@legal-assistant |
| CI/CD Maintainer | CI Bot | ci-bot@github |
| Infrastructure Owner | Alex | [alex@legal-assistant.tech](mailto:alex@legal-assistant.tech) |
| Telegram Channel | @legal\_assistant\_bot | уведомления CI/CD |

## ✅ Заключение

**Legal Assistant Arbitrage v2.4** — это зрелая и надёжная DevOps-инфраструктура, где CI/CD, тестирование, деплой и мониторинг связаны в единый процесс.

* Автоматизированный пайплайн GitHub Actions
* Полная тестовая интеграция (unit, smoke, integration)
* Telegram-уведомления о всех событиях
* Документированные ручные и аварийные сценарии
* Полный журнал релизов и rollback-ов

📅 *Обновлено:* 2025-10-09 📦 *Версия руководства:* v2.4.1 🧾 *Утверждено:* DevOps Department — Legal Assistant Arbitrage

---  
  
📘 Готово — это \*\*единый PDF-ready документ\*\*, который объединяет всю DevOps-документацию.   
  
Хочешь, чтобы я сгенерировал из него готовый PDF-файл с титульным листом, логотипом ⚖️ и автоматическим оглавлением?