

Что проверить перед релизом

Неприятно получить HTTP 500 Server Error, реализовав крутую фичу, пройдя долгий путь тестирования и выпуска релиза, потому что забыли о настройке

2 страницы настроек, без которых фича может не запуститься или быстро сломаться

ДО ПЕРВОГО РЕЛИЗА

☐ Режим отладки выключен [DEBUG = False](#)

Не нужно раскрывать из чего состоит приложение, как настроено, чтобы не быть автоматически взломанным.

☐ Настройки не хранятся в коде

Читай настройки из переменных окружения при помощи [django-environ](#) и шифруй их при помощи [sops](#), если хочешь хранить в репозитории

☐ SECRET_KEY обновлен и разный для local/prod

Зная SECRET_KEY можно расшифровать пароли. Обнови его при помощи Django или [Djecrety](#) и читай из переменных окружения.

☐ Настроен список клиентов в ALLOWED_HOSTS

Не выходи в интернет без базовой защиты, не обслуживай тех, кому не доверяешь. Перечисли [свои домены](#) и свои [IP-подсети](#)

☐ Настроена раздача статических файлов

Сохрани и настрой URL для артефактов (особенно .css, .js), иначе удивись пустым страницам.

Укажи [STATIC_URL, STATIC_ROOT](#), а потом освой Nginx

☐ Проверена готовность к релизу [check --deploy](#)

Убедись, что ничего лишнего не попало в репозиторий, зависимости актуализированы, тесты проходят



ДО ПЕРВОГО ПОЛЬЗОВАТЕЛЯ

☐ Настроено подключение к базе данных

Укажи не только путь до базы, пользователя и пароль, но еще и [CONN_MAX_AGE](#), [CONNECTION_POOL](#)

☐ Ошибки и логи собираются

Узнавай об ошибках одновременно с пользователем при помощи [LOGGING](#), [django-structlog](#) и [Sentry/BugSnag](#)

☐ Используется UWSGI/Gunicorn

Чтобы обрабатывать запросы больше и лучше, настрой [UWSGI](#) или [Gunicorn](#). Они сами перезапустят Django, если “зависнет”

☐ Работает по HTTPS и защищен от простых атак

Настроен [SSL сертификат](#) и включены все [SECURE](#) * настройки для перенаправления на HTTPS, защиты COOKIE

ДО ПЕРВОЙ ОПЛАТЫ

☐ Делаются и проверяются резервные копии

Разберись как делать бэкапы при помощи [django-dbbackup](#) или [pg_dump](#) (PostgreSQL) и регулярно проверяй, что они работают 🧐

☐ Автоматическая сборка и релиз приложения

Запускай тесты, собирай [Docker образ](#) и делай релиз без применения твоей рабочей станции, освой CI.

☐ Определяется статус приложения по метрикам

Настрой [django-health-check](#), чтобы узнавать о поломке раньше пользователя. Собирай метрики приложения с [django-prometheus](#)

☐ Страницы грузятся быстрее 200ms

Сжимай размер ответа при помощи [GZIP](#) и [django-brotli](#), настрой [CACHE](#) для дополнительного ускорения своих фич.