

Чеклист для развертывания WOWsilizing Bot

Перед началом развертывания

Шаг 1: Подготовка учетных записей

Railway аккаунт

- ☐ Зарегистрирован на <https://railway.app/>
- ☐ Email подтвержден
- ☐ (Опционально) Добавлена банковская карта для увеличения лимитов
- ☐ Понимание тарифов: \$5 бесплатно/месяц

Telegram Bot

- ☐ Бот создан через @BotFather
- ☐ Токен бота получен: `8314895069:AAG1P9oozB0Hv1pMaIPy-uzGQhayu6Fz9c8`
- ☐ Username бота: @WOWsilizing
- ☐ Premium username: @WowFUX

GitHub (рекомендуется)

- ☐ Аккаунт GitHub создан
 - ☐ Репозиторий создан (или готов к созданию)
 - ☐ Код загружен в репозиторий
-

Шаг 2: API ключи для AI функций (ОПЦИОНАЛЬНО)

OpenAI (для Whisper, GPT-4, TTS)

- ☐ Аккаунт создан на <https://platform.openai.com/>
- ☐ API ключ получен
- ☐ Баланс пополнен (\$5+ рекомендуется)
- ☐ Установлены лимиты расходов
- ☐ Ключ скопирован в безопасное место

Стоимость: ~\$0.20 за 10-минутное видео

Google Generative AI (для Gemini, TTS)

- ☐ Аккаунт создан на <https://makersuite.google.com/>
- ☐ API ключ получен
- ☐ Понимание лимитов: бесплатно до определенного использования
- ☐ Ключ скопирован

Стоимость: Бесплатно с ограничениями

ElevenLabs (для премиум TTS)

- ☐ Аккаунт создан на <https://elevenlabs.io/>
- ☐ API ключ получен

- ☐ Выбран тариф (Free/Starter/Creator/Pro)
- ☐ Ключ скопирован

Стоимость: Free: 10,000 символов/месяц

Шаг 3: Подготовка кода

✓ Файлы проекта

- ☐ `bot.py` - основной файл бота
- ☐ `config.py` - конфигурация
- ☐ `database.py` - работа с БД
- ☐ `ai_processor.py` - AI функции
- ☐ `video_processor.py` - обработка видео
- ☐ `utils.py` - утилиты
- ☐ `requirements.txt` - зависимости
- ☐ `Dockerfile` - конфигурация Docker
- ☐ `.dockerignore` - исключения для Docker
- ☐ `railway.json` - конфигурация Railway
- ☐ `.env.example` - пример переменных окружения
- ☐ `README.md` - документация
- ☐ `DEPLOYMENT.md` - руководство по разворачиванию
- ☐ `QUICK_START.md` - быстрый старт
- ☐ `ENV_VARIABLES.md` - описание переменных
- ☐ `CHECKLIST.md` - этот файл
- ☐ `POST_DEPLOYMENT.md` - пост-разворачивание

✓ Проверка кода

- ☐ Все зависимости указаны в `requirements.txt`
 - ☐ `Dockerfile` корректен
 - ☐ Нет хардкода токенов/ключей в коде
 - ☐ `.env` добавлен в `.gitignore`
 - ☐ Код протестирован локально (если возможно)
-

Развертывание на Railway

Шаг 4: Загрузка кода

Метод 1: Через GitHub (рекомендуется)

- ☐ Код закоммичен в git
- ☐ Репозиторий создан на GitHub
- ☐ Код запущен на GitHub
- ☐ Репозиторий доступен (приватный или публичный)

Метод 2: Прямая загрузка

- ☐ ZIP архив создан со всеми необходимыми файлами

- [] Исключены ненужные файлы (temp/, data/, logs/, **pycache/**)
 - [] Архив готов к загрузке
-

Шаг 5: Создание проекта в Railway

- [] Вошли в Railway Dashboard
 - [] Нажали “New Project”
 - [] Выбрали метод развертывания (GitHub/Upload)
 - [] Репозиторий/файлы загружены
 - [] Сборка началась
 - [] Сборка завершилась успешно (статус “Success”)
-

Шаг 6: Настройка переменных окружения

✓ ОБЯЗАТЕЛЬНЫЕ переменные

- [] `BOT_TOKEN` добавлен (ваш токен бота)
- [] `PREMIUM_USERNAME` добавлен (WowFUX без @)
- [] `LOG_LEVEL` установлен (INFO рекомендуется)

✓ ОПЦИОНАЛЬНЫЕ переменные (для AI)

- [] `OPENAI_API_KEY` добавлен (если нужен)
- [] `GOOGLE_API_KEY` добавлен (если нужен)
- [] `ELEVENLABS_API_KEY` добавлен (если нужен)

✓ Проверка

- [] Все переменные без лишних пробелов
 - [] Токены скопированы правильно
 - [] Бот автоматически перезапустился после добавления переменных
-

Шаг 7: Проверка развертывания

✓ Статус в Railway

- [] Статус проекта “Active” (зеленый)
- [] Нет ошибок в логах
- [] Видно сообщение “Bot started successfully”
- [] Видно сообщение “Started polling”

✓ Тест в Telegram

- [] Бот найден по username @WOWsilizing
- [] Отправили команду `/start`
- [] Бот ответил приветственным сообщением
- [] Меню команд отображается
- [] Кнопки работают

✓ Функциональный тест

- ☐ Отправили видео или YouTube ссылку
 - ☐ Бот принял файл
 - ☐ Показаны кнопки действий
 - ☐ Одна из бесплатных функций работает (например, извлечение аудио)
 - ☐ Файл обработан и отправлен обратно
-



Мониторинг и настройка

Шаг 8: Настройка мониторинга

- ☐ Проверен Usage в Railway Dashboard
- ☐ Понимание текущих расходов
- ☐ (Опционально) Добавлены уведомления о превышении лимитов
- ☐ Настроен мониторинг логов

Шаг 9: Оптимизация

- ☐ Проверены метрики (CPU, Memory, Disk)
 - ☐ Все в пределах нормы
 - ☐ Кеш работает корректно
 - ☐ Временные файлы удаляются
-



Безопасность

Шаг 10: Проверка безопасности

✓ Токены и ключи

- ☐ Токены НЕ опубликованы в открытом доступе
- ☐ `.env` добавлен в `.gitignore`
- ☐ Ключи хранятся только в Railway Variables
- ☐ Нет хардкода ключей в коде
- ☐ Используются переменные окружения

✓ Лимиты расходов

- ☐ OpenAI: установлены лимиты расходов
- ☐ Google: проверены квоты
- ☐ ElevenLabs: установлены spending caps
- ☐ Railway: понимание биллинга

✓ Доступ

- ☐ Только авторизованные пользователи имеют доступ к Railway проекту
 - ☐ Премиум функции доступны только @WowFUX
 - ☐ Обычные пользователи имеют доступ только к бесплатным функциям
-

Документация

Шаг 11: Изучение документации

- ☐ Прочитан DEPLOYMENT.md
 - ☐ Прочитан QUICK_START.md
 - ☐ Прочитан ENV_VARIABLES.md
 - ☐ Прочитан POST_DEPLOYMENT.md
 - ☐ Понятна структура проекта
 - ☐ Известны основные команды бота
-

Готовность к запуску

Финальная проверка

Базовая функциональность

- ☐ Бот запущен и работает 24/7
- ☐ Отвечает на команды
- ☐ Обрабатывает видео
- ☐ Бесплатные функции работают
- ☐ Нет критических ошибок в логах

AI функции (если настроены)

- ☐ Whisper транскрипция работает
- ☐ GPT-4 перевод работает
- ☐ TTS озвучка работает
- ☐ Выбран предпочтительный TTS сервис
- ☐ API ключи валидны


Производительность

- ☐ CPU использование < 50% в среднем
- ☐ Memory использование < 500MB
- ☐ Disk использование разумное
- ☐ Время обработки приемлемое

Биллинг

- ☐ Railway кредиты достаточны
 - ☐ OpenAI баланс положительный (если используется)
 - ☐ Понимание текущих расходов
 - ☐ Установлены уведомления о лимитах
-

Готово к производству!

Если все пункты выше отмечены , ваш бот готов к использованию в продакшене!

Что делать при проблемах?

Бот не запускается:

1. Проверьте логи в Railway
2. Убедитесь, что BOT_TOKEN правильный
3. Проверьте все переменные окружения
4. Попробуйте Redeploy

Бот не отвечает:

1. Проверьте статус в Railway (должен быть Active)
2. Проверьте логи на наличие ошибок
3. Убедитесь, что токен не отозван
4. Попробуйте перезапустить бота

AI функции не работают:

1. Проверьте валидность API ключей
2. Проверьте баланс на AI сервисах
3. Убедитесь, что пользователь имеет премиум доступ (@WowFUX)
4. Проверьте логи на специфические ошибки AI

Превышен лимит кредитов:




1. Проверьте Usage в Railway Dashboard
2. Оптимизируйте код (кеширование)
3. Добавьте банковскую карту
4. Перейдите на платный план

Полезные ссылки



Railway

-  Dashboard: <https://railway.app/dashboard>
-  Документация: <https://docs.railway.app/>
-  Discord: <https://discord.gg/railway>
-  Status: <https://status.railway.app/>

AI Сервисы

-  OpenAI Platform: <https://platform.openai.com/>
-  Google AI: <https://makersuite.google.com/>
-  ElevenLabs: <https://elevenlabs.io/>

Telegram

-  BotFather: <https://t.me/BotFather>
-  Ваш бот: <https://t.me/WOWsilizing>

Документация проекта

-  Полное руководство: [DEPLOYMENT.md](#) (DEPLOYMENT.md)
-  Быстрый старт: [QUICK_START.md](#) (QUICK_START.md)

- 🔑 Переменные окружения: [ENV_VARIABLES.md](#) (ENV_VARIABLES.md)
 - 📊 Пост-развертывание: [POST_DEPLOYMENT.md](#) (POST_DEPLOYMENT.md)
 - 📄 README: [README.md](#) (README.md)
-

Поздравляем!

Если вы прошли все пункты чеклиста - ваш бот WOWsilizing успешно развернут и готов к работе!

Следующие шаги:

1. ☒ Протестируйте все функции
2. ☒ Пригласите пользователей
3. ☒ Следите за логами и метриками
4. ☒ Оптимизируйте при необходимости
5. ☒ Добавляйте новые функции

Удачи с вашим ботом! 🚀

Последнее обновление: Ноябрь 2024

Версия: 1.0