



# Чек-лист развертывания WOWsilizing Bot



Используйте этот чек-лист, чтобы убедиться, что ничего не упустили!



Отмечайте пункты по мере выполнения: ☐ → ☒



## Часть 1: Подготовка (ДО развертывания)



### Telegram бот

- ☐ Создал бота через @BotFather
- ☐ Получил BOT\_TOKEN
- ☐ Записал BOT\_TOKEN в надежное место
- ☐ Настроил имя бота (понятное пользователям)
- ☐ Настроил описание бота
- ☐ Добавил аватар боту (опционально)
- ☐ Установил команды через /setcommands (опционально)

#### BOT\_TOKEN:

(Запишите здесь для удобства, но НЕ загружайте этот файл в публичный репозиторий!)



### Подготовка файлов

- ☐ Все файлы бота на месте в одной папке
- ☐ Проверил наличие основных файлов:
- ☐ `bot.py`
- ☐ `config.py`
- ☐ `database.py`
- ☐ `video_processor.py`
- ☐ `ai_processor.py`
- ☐ `utils.py`
- ☐ `requirements.txt`
- ☐ `Dockerfile`
- ☐ `README.md`
- ☐ `.gitignore`
- ☐ `.env.example`
- ☐ Создал `.env` файл локально (для тестирования)
- ☐ Проверил, что `.env` в `.gitignore` (НЕ должен загружаться на GitHub!)
- ☐ Проверил, что нет лишних файлов (кеш, временные)

## API ключи (опционально для AI функций)

Если хотите использовать AI:

### OpenAI (для транскрипции и TTS)

- ☐ Зарегистрировался на <https://platform.openai.com>
- ☐ Создал API ключ
- ☐ Добавил средства на счет (\$5 минимум)
- ☐ Записал OPENAI\_API\_KEY

**OPENAI\_API\_KEY:**

### Google AI (альтернатива OpenAI)

- ☐ Зарегистрировался на <https://makersuite.google.com>
- ☐ Создал API ключ
- ☐ Записал GOOGLE\_AI\_API\_KEY

**GOOGLE\_AI\_API\_KEY:**

### ElevenLabs (для качественного TTS)

- ☐ Зарегистрировался на <https://elevenlabs.io>
- ☐ Создал API ключ
- ☐ Записал ELEVENLABS\_API\_KEY

**ELEVENLABS\_API\_KEY:**

---

## Часть 2: GitHub

### Создание аккаунта (если нет)

- ☐ Зарегистрировался на <https://github.com>
- ☐ Подтвердил email
- ☐ Выбрал бесплатный план (Free)
- ☐ Настроил профиль (опционально)

**GitHub Username:**

## Создание репозитория

- ☐ Создал новый репозиторий
- ☐ Назвал репозиторий (например: `wowsilizing-bot` )
- ☐ Выбрал Public (или Private, если нужно)
- ☐ НЕ добавлял README, .gitignore, LICENSE (они уже есть)
- ☐ Записал адрес репозитория

### Адрес репозитория:

```
https://github.com/_____/_____  
                   (ваш username)  (название репо)
```

## Загрузка кода

Выберите ОДИН метод и отметьте:

### Метод 1: Веб-интерфейс (самый простой)

- ☐ Открыл репозиторий на GitHub
- ☐ Нажал "Upload files" или "uploading an existing file"
- ☐ Перетащил ВСЕ файлы бота
- ☐ Написал commit message: "Первая загрузка бота"
- ☐ Нажал "Commit changes"
- ☐ Проверил, что все файлы загрузились
- ☐ Убедился, что `.env` НЕ загрузился (должен быть только `.env.example` )

### Метод 2: GitHub Desktop

- ☐ Скачал и установил GitHub Desktop
- ☐ Вошел в аккаунт GitHub
- ☐ Клонировал репозиторий
- ☐ Скопировал файлы бота в папку репозитория
- ☐ Создал коммит: "Первая загрузка бота"
- ☐ Нажал "Push origin"
- ☐ Проверил на GitHub, что файлы загрузились

### Метод 3: Git командная строка

- ☐ Установил Git (если не был установлен)
- ☐ Настроил Git (user.name и user.email)
- ☐ Открыл командную строку в папке бота
- ☐ Выполнил `git init`
- ☐ Выполнил `git add .`
- ☐ Выполнил `git commit -m "Первая загрузка бота"`
- ☐ Выполнил `git remote add origin https://github.com/...`
- ☐ Выполнил `git branch -M main`
- ☐ Выполнил `git push -u origin main`
- ☐ Проверил на GitHub, что файлы загрузились

## Проверка репозитория

- ☐ Все файлы видны на GitHub
  - ☐ `.env` файла НЕТ на GitHub (только `.env.example` )
  - ☐ Файлы открываются и отображаются правильно
  - ☐ README.md отображается на главной странице
- 



## Часть 3: Railway

---

### Создание аккаунта

- ☐ Открыл <https://railway.app>
  - ☐ Нажал “Login”
  - ☐ Вошел через GitHub (“Login with GitHub”)
  - ☐ Разрешил Railway доступ к GitHub
  - ☐ Подтвердил email
- 

### Добавление способа оплаты

- ☐ Начал пробный период (“Start Trial”)
  - ☐ Добавил кредитную/дебетовую карту
  - ☐ Ввел данные карты и адрес
  - ☐ Подтвердил (проверка \$1)
  - ☐ Получил \$5 бесплатных кредитов
  - ☐ Проверил, что Trial активен
- 

### Создание проекта

- ☐ Нажал “New Project”
  - ☐ Выбрал “Deploy from GitHub repo”
  - ☐ Нажал “Configure GitHub App” (если первый раз)
  - ☐ Дал доступ Railway к репозиторию (выбрал `wowsilizing-bot` )
  - ☐ Вернулся в Railway
  - ☐ Выбрал репозиторий `wowsilizing-bot`
  - ☐ Нажал “Deploy Now”
  - ☐ Дождался создания проекта
- 

### Настройка переменных окружения

- ☐ Открыл карточку сервиса
- ☐ Перешел на вкладку “Variables”

**Обязательные переменные:****BOT\_TOKEN:**

- [ ] Добавил переменную `BOT_TOKEN`
- [ ] Вставил токен от BotFather
- [ ] Проверил, что токен скопирован полностью
- [ ] Нажал “Add”

**PREMIUM\_USERNAME:**

- [ ] Добавил переменную `PREMIUM_USERNAME`
- [ ] Вставил свой Telegram username (БЕЗ @)
- [ ] Нажал “Add”

**Опциональные переменные (если используете AI):****OPENAI\_API\_KEY** (если есть):

- [ ] Добавил переменную `OPENAI_API_KEY`
- [ ] Вставил ключ
- [ ] Нажал “Add”

**GOOGLE\_AI\_API\_KEY** (если есть):

- [ ] Добавил переменную `GOOGLE_AI_API_KEY`
- [ ] Вставил ключ
- [ ] Нажал “Add”

**ELEVENLABS\_API\_KEY** (если есть):

- [ ] Добавил переменную `ELEVENLABS_API_KEY`
- [ ] Вставил ключ
- [ ] Нажал “Add”

**LOG\_LEVEL** (опционально):





- [ ] Добавил переменную `LOG_LEVEL`
- [ ] Установил значение `INFO`
- [ ] Нажал “Add”

**Запуск развертывания**

- [ ] Railway автоматически начал развертывание (или нажал “Deploy”)
- [ ] Перешел на вкладку “Deployments”
- [ ] Открыл активное развертывание
- [ ] Наблюдаю за логами в реальном времени

**Часть 4: Ожидание развертывания****Мониторинг процесса**


- [ ] Вижу “Cloning repository...”
- [ ] Вижу “Building Docker image...”
- [ ] Вижу “Installing system dependencies...”
- [ ] Вижу “Installing Python packages...”

- ☐ Вижу “Starting bot...” 
- ☐ Вижу “Bot started successfully” 
- ☐ Вижу “Started polling” 
- ☐ Статус изменился на  Active

**Время развертывания:** 3-5 минут

---

## Проверка ошибок

Если развертывание упало ( Crashed):


- ☐ Открыл логи
- ☐ Нашел строку с “Error:”
- ☐ Записал ошибку:

**Ошибка:**

- ☐ Проверил раздел “Решение проблем” в RAILWAY\_GITHUB\_DEPLOY.md
  - ☐ Исправил проблему
  - ☐ Перезапустил развертывание
- 

## Часть 5: Проверка работы

### Проверка в Railway


- ☐ Статус сервиса:  Active
  - ☐ В логах вижу “Bot started successfully”
  - ☐ В логах нет ошибок (Error:)
  - ☐ CPU Usage отображается
  - ☐ Memory Usage отображается
- 

### Проверка в Telegram

**Базовые проверки:**

- ☐ Открыл бота в Telegram
- ☐ Бот онлайн (зеленая точка рядом с именем)
- ☐ Отправил `/start`
- ☐ Бот ответил приветственным сообщением
- ☐ Вижу меню с кнопками
- ☐ Отправил `/help`
- ☐ Получил список команд

### Проверка обработки видео:

- ☐ Отправил короткое видео боту
- ☐ Бот принял видео
- ☐ Бот показал кнопки действий
- ☐ Нажал  "Вырезать фрагмент"
- ☐ Ввел тайм-код (например: 00:00:05-00:00:10 )
- ☐ Бот начал обработку
- ☐ Бот отправил обработанное видео
- ☐ Видео воспроизводится корректно

### Проверка других функций:

- ☐ /settings - работает
- ☐ /templates - работает (если есть шаблоны)
- ☐ Извлечение аудио - работает
- ☐ Обнаружение тишины - работает (если премиум)
- ☐ Другие функции - работают

### Проверка премиум функций (если настроены AI ключи)

- ☐ Транскрипция аудио в текст
- ☐ Перевод субтитров
- ☐ Синтез речи (TTS)
- ☐ Автоопределение важных моментов
- ☐ Суммаризация видео



## Часть 6: Мониторинг

### Настройка мониторинга

- ☐ Открыл вкладку "Metrics" в Railway
- ☐ Проверил CPU Usage (должно быть <50% в покое)
- ☐ Проверил Memory Usage (должно быть <200MB в покое)
- ☐ Проверил Disk Usage (должно быть минимальное)

### Настройка уведомлений

- ☐ Перешел в Settings → Notifications
- ☐ Добавил свой email
- ☐ Включил уведомления о:
- ☐ Падении сервиса (Service down)
- ☐ Превышении лимитов (Limits exceeded)
- ☐ Завершении развертывания (Deployment complete)

## Проверка использования

- ☐ Перешел в Settings → Usage
- ☐ Проверил текущие расходы
- ☐ Проверил использованные часы
- ☐ Установил лимит расходов (опционально):

**Мой лимит:** \$\_ в месяц

---



## Часть 7: Документация

---

### Личная документация

- ☐ Создал файл с важной информацией:
  - ☐ Адрес репозитория GitHub
  - ☐ Название проекта в Railway
  - ☐ Имя бота в Telegram
  - ☐ BOT\_TOKEN (в безопасном месте!)
  - ☐ API ключи (в безопасном месте!)
  - ☐ Дата развертывания
  - ☐ Контакты поддержки
- 

### Обновление README

- ☐ Обновил README.md с информацией о боте
  - ☐ Добавил описание функций
  - ☐ Добавил инструкции для пользователей
  - ☐ Добавил ссылку на бота (опционально)
- 



## Часть 8: Финализация

---

### Тестирование с реальными пользователями

- ☐ Поделится ботом с 2-3 друзьями
  - ☐ Попросил протестировать основные функции
  - ☐ Собрал обратную связь
  - ☐ Исправил найденные проблемы
- 

### Оптимизация

- ☐ Проверил логи на предупреждения (Warning:)
- ☐ Оптимизировал медленные функции (если есть)
- ☐ Настроил автоочистку временных файлов
- ☐ Проверил, что база данных не растет слишком быстро



---

## Безопасность

- ☐ Убедился, что `.env` НЕ на GitHub
  - ☐ Проверил, что токены нигде не засвечены
  - ☐ Настроил ротацию токенов (запланировал на будущее)
  - ☐ Добавил двухфакторную аутентификацию на GitHub (опционально)
  - ☐ Добавил двухфакторную аутентификацию на Railway (опционально)
- 

## Резервное копирование

- ☐ Сделал бэкап базы данных (если есть важные данные)
  - ☐ Сохранил копию всех настроек
  - ☐ Записал все переменные окружения в безопасное место
  - ☐ Создал план восстановления (что делать если что-то сломается)
- 



## Часть 9: План на будущее

---

### Краткосрочные задачи (первая неделя)

- ☐ Мониторить работу бота ежедневно
  - ☐ Проверять логи на ошибки
  - ☐ Следить за расходами
  - ☐ Собирать обратную связь
  - ☐ Исправлять критичные баги
- 

### Среднесрочные задачи (первый месяц)

- ☐ Добавить новые функции (если нужно)
  - ☐ Оптимизировать производительность
  - ☐ Написать документацию для пользователей
  - ☐ Настроить автоматические бэкапы
  - ☐ Изучить аналитику использования
- 

### Долгосрочные задачи (3-6 месяцев)

- ☐ Масштабировать, если много пользователей
  - ☐ Добавить продвинутые AI функции
  - ☐ Создать landing page для бота
  - ☐ Настроить мониторинг и алерты
  - ☐ Рассмотреть монетизацию (если актуально)
-

## ✓ ИТОГОВАЯ ПРОВЕРКА

### Критические пункты (должны быть ВСЕ отмечены!):

- [ ] ✓ Бот работает и отвечает в Telegram
- [ ] ✓ Обработка видео работает корректно
- [ ] ✓ Логи показывают "Bot started successfully"
- [ ] ✓ Статус в Railway: ● Active
- [ ] ✓ .env файл НЕ загружен на GitHub
- [ ] ✓ Все токены и ключи в безопасности
- [ ] ✓ Настроены уведомления о проблемах
- [ ] ✓ Понимаю, как перезапустить бота
- [ ] ✓ Понимаю, как обновить код
- [ ] ✓ Знаю, куда обращаться за помощью

## 🎯 Следующие шаги

### Если ВСЕ критические пункты отмечены:

#### ✓ ПОЗДРАВЛЯЮ! БОТ УСПЕШНО РАЗВЕРНУТ! 🎉

Теперь:

1. Наслаждайтесь работой бота
2. Делитесь с друзьями
3. Собирайте обратную связь
4. Улучшайте и развивайте

### Если есть проблемы:

✗ Откройте соответствующий раздел:

- RAILWAY\_GITHUB\_DEPLOY.md → Раздел "Решение проблем"
- GITHUB\_SETUP.md → Раздел "Частые проблемы"
- QUICK\_REFERENCE.md → Быстрая справка

## 📖 Полезные файлы

- 📖 GITHUB\_SETUP.md - Подробная инструкция по GitHub
- 🚀 RAILWAY\_GITHUB\_DEPLOY.md - Подробная инструкция по Railway
- 📄 COMPLETE\_GUIDE.md - Полное руководство от А до Я
- 🚀 QUICK\_REFERENCE.md - Быстрая справка
- 🎬 VIDEO\_SCRIPT.md - Текстовая инструкция
- 📄 README.md - Описание бота
- 📁 FILES\_MANIFEST.md - Описание всех файлов



## Советы

---

1. **Распечатайте этот чек-лист** или откройте в отдельной вкладке
  2. **Отмечайте пункты по мере выполнения** - так не запутаетесь
  3. **Не торопитесь** - лучше потратить больше времени, но сделать правильно
  4. **Сохраните все токены и ключи** в надежном месте (менеджер паролей)
  5. **Делайте скриншоты** важных моментов - пригодятся при проблемах
- 



## Статистика развертывания

---

Дата начала: \_\_\_\_\_

Дата завершения: \_\_\_\_\_

Время потрачено: \_\_\_\_\_ минут

Проблем возникло: \_\_\_\_\_

Проблем решено: \_\_\_\_\_

Оценка сложности (1-10): \_\_\_\_\_

Комментарии:

---

---

---

---

💬 Поздравляем с успешным развертыванием! 🚀

Теперь ваш бот работает 24/7 и готов обрабатывать видео!