

Полное руководство по развертыванию **WOWsilizing** бота

Содержание

1. [Введение](#)
2. [Что вам понадобится](#)
3. [Шаг 1: Создание аккаунта на GitHub](#)
4. [Шаг 2: Создание репозитория на GitHub](#)
5. [Шаг 3: Загрузка кода бота](#)
6. [Шаг 4: Создание аккаунта на Railway](#)
7. [Шаг 5: Подключение GitHub к Railway](#)
8. [Шаг 6: Настройка переменных окружения](#)
9. [Шаг 7: Запуск и тестирование](#)
10. [Устранение неполадок](#)

Введение

Это пошаговое руководство поможет вам развернуть Telegram бота **WOWsilizing** на платформе Railway через GitHub. Даже если вы никогда не работали с GitHub или хостингом, следуя этим инструкциям, вы сможете запустить бота за **30-40 минут**.

Почему GitHub + Railway?

-  **Независимость:** Ваш код хранится на GitHub, вы полностью контролируете его
-  **Бесплатно:** GitHub бесплатен, Railway дает \$5 кредитов каждый месяц
-  **Просто:** Не нужно знать программирование или командную строку
-  **Автообновление:** Изменения в GitHub автоматически применяются на Railway

Что вам понадобится

Перед началом подготовьте:

- [] **Email адрес** (для регистрации на GitHub и Railway)
- [] **Токен Telegram бота** (получите у [@BotFather](https://t.me/BotFather) (<https://t.me/BotFather>))
- [] **Username премиум пользователя** (например: `WowFUX`)
- [] **Файлы бота** (все файлы из папки `/home/ubuntu/wowsilizing_bot/`)
- [] **30-40 минут свободного времени** ☕

Опционально (для премиум функций):

- [] OpenAI API ключ (для Whisper, GPT-4, TTS)
- [] Google API ключ (для Gemini)

- [] ElevenLabs API ключ (для голосовой озвучки)

 **Совет:** Даже без AI ключей бот будет работать с базовыми функциями!

Шаг 1: Создание аккаунта на GitHub

 **Время: 5 минут**

GitHub — это платформа для хранения кода. Здесь будет храниться код вашего бота.

1.1 Регистрация

1.  Откройте в браузере: <https://github.com/signup>
2.  Введите ваш **email адрес** и нажмите **Continue**
3.  Придумайте **пароль** (минимум 15 символов или 8 символов с цифрой и буквой)
- Пример хорошего пароля: `MyBot2024Secure!`
4.  Придумайте **username** (имя пользователя)
- Это будет ваш адрес: `github.com/ваш-username`
- Можно использовать латинские буквы, цифры и дефис
- Пример: `wowsilizing-bot` или `myawesomebot`
5.  Выберите, хотите ли получать новости от GitHub (можно отказаться)
6.  Пройдите проверку (нажмите кнопку или решите головоломку)
7.  Откройте вашу почту и найдите письмо от GitHub
8.  Введите **код подтверждения** из письма
9.  Поздравляем! Аккаунт создан!

1.2 Настройка профиля (опционально)

После входа GitHub может задать несколько вопросов:

- Сколько человек в вашей команде? → Выберите **Just me**
- Вы студент или преподаватель? → Выберите подходящий вариант или пропустите
- Что вас интересует? → Можете пропустить

Нажмите **Skip** или **Complete setup** внизу страницы.

Шаг 2: Создание репозитория на GitHub

 **Время: 3 минуты**

Репозиторий — это папка для вашего проекта на GitHub.

2.1 Создание нового репозитория

1.  На главной странице GitHub нажмите зеленую кнопку **New** (слева вверху)
- Или перейдите по ссылке: <https://github.com/new>

2. Заполните форму:

Repository name (Имя репозитория):

wowsilizing-bot

Можете выбрать любое имя, но используйте только латинские буквы, цифры и дефисы

Description (Описание) — optional:

Telegram бот для обработки видео с AI функциями

Public или Private:

- Выберите **Public** (публичный) — бесплатно и удобно
- Или **Private** (приватный) — если хотите скрыть код (тоже бесплатно)

Initialize this repository with:

- Поставьте галочку **Add a README file**
- Остальные галочки не ставьте

1. Нажмите зеленую кнопку **Create repository** внизу

2. Готово! Вы создали репозиторий!

Теперь вы увидите страницу вашего репозитория с адресом:

<https://github.com/ ваш-username /wowsilizing-bot>



Шаг 3: Загрузка кода бота

Время: 10-15 минут

Теперь загрузим все файлы бота в репозиторий. Мы используем **веб-интерфейс** — самый простой способ без командной строки!

3.1 Подготовка файлов

1. Откройте папку с файлами бота на вашем компьютере
 - Путь: /home/ubuntu/wowsilizing_bot/
2. Убедитесь, что у вас есть эти **обязательные файлы**:

<input checked="" type="checkbox"/> bot.py	(главный файл бота)
<input checked="" type="checkbox"/> ai_processor.py	(обработка AI)
<input checked="" type="checkbox"/> database.py	(база данных)
<input checked="" type="checkbox"/> utils.py	(вспомогательные функции)
<input checked="" type="checkbox"/> video_processor.py	(обработка видео)
<input checked="" type="checkbox"/> config.py	(конфигурация)
<input checked="" type="checkbox"/> requirements.txt	(зависимости Python)
<input checked="" type="checkbox"/> Dockerfile	(для Railway)
<input checked="" type="checkbox"/> railway.json	(настройки Railway)
<input checked="" type="checkbox"/> .env.example	(пример переменных)
<input checked="" type="checkbox"/> .gitignore	(игнорируемые файлы)
<input checked="" type="checkbox"/> .dockerignore	(игнорируемые при сборке)

3. **⚠ ВАЖНО:** НЕ загружайте файлы:
 - `.env` (если есть) — содержит секретные ключи!
 - `*.db` — файлы базы данных
 - `__pycache__/` — временные файлы Python
 - Любые PDF файлы документации (они большие)

3.2 Загрузка через веб-интерфейс

Метод 1: Загрузка по одному файлу (проще)

1. Откройте страницу вашего репозитория:
<https://github.com/ваш-username/wowsilizing-bot>
2. Нажмите кнопку **Add file → Upload files**
3. Перетащите файлы в окно браузера или нажмите **choose your files**
4. Выберите **все файлы из списка выше** (кроме исключений)
 - Можно выбрать несколько файлов, удерживая Ctrl (Windows) или Cmd (Mac)
5. Внизу страницы в поле **Commit changes** напишите:
 Загрузка кода бота
6. Нажмите зеленую кнопку **Commit changes**
7. Подождите, пока файлы загружаются (может занять 1-2 минуты)

Метод 2: Создание файлов вручную (если загрузка не работает)

Если у вас проблемы с загрузкой, можно создать файлы вручную:

1. На странице репозитория нажмите **Add file → Create new file**
2. В поле **Name your file** введите имя файла, например: `bot.py`
3. Скопируйте содержимое файла с вашего компьютера и вставьте в большое текстовое поле
4. Внизу нажмите **Commit new file**
5. Повторите для каждого файла из списка

Совет: Этот метод дольше, но надежнее для больших файлов

3.3 Проверка загрузки

После загрузки на главной странице репозитория вы должны увидеть все файлы:

```
wowsilizing-bot
├── .dockerignore
├── .env.example
├── .gitignore
├── Dockerfile
├── README.md
├── ai_processor.py
├── bot.py
├── config.py
├── database.py
├── railway.json
├── requirements.txt
├── utils.py
└── video_processor.py
```

- Если все файлы на месте — отлично, переходим дальше!

Шаг 4: Создание аккаунта на Railway

Время: 5 минут

Railway — это платформа для хостинга приложений. Здесь будет работать ваш бот 24/7.

4.1 Регистрация

1. Откройте в браузере: <https://railway.app/>
2. Нажмите кнопку **Start a New Project** или **Login**
3. Выберите способ входа:
 - **GitHub** (рекомендуется) — войдите через ваш GitHub аккаунт
 - Или **Email** — введите email и пароль

4.2 Вход через GitHub (рекомендуется)

1. Нажмите **Login with GitHub**
2. GitHub попросит разрешение для Railway:
 - Прочтайте, какие права запрашивает Railway
 - Нажмите зеленую кнопку **Authorize Railway**
3. Вы автоматически войдете в Railway!

4.3 Подтверждение email

1. Railway отправит письмо на ваш email
2. Откройте письмо и нажмите **Verify Email**
3. Email подтвержден!

4.4 Бесплатный план

Railway дает **\$5 кредитов каждый месяц** бесплатно!

Для Telegram бота этого обычно достаточно. Если кредиты закончатся, Railway попросит добавить карту, но **не спишет деньги без вашего разрешения**.

 **Совет:** Следите за использованием кредитов в разделе **Usage** на Railway

Шаг 5: Подключение GitHub к Railway

 **Время: 5 минут**

Теперь свяжем ваш репозиторий на GitHub с Railway для автоматического развертывания.

5.1 Создание нового проекта

1.  На главной странице Railway нажмите **New Project**
2.  Выберите **Deploy from GitHub repo**
3.  Если это первый раз, Railway попросит доступ к вашим репозиториям:
 - Нажмите **Configure GitHub App**
 - Выберите **Only select repositories**
 - Найдите и выберите `wowsilizing-bot`
 - Нажмите **Install & Authorize**
4.  Вернитесь на Railway и выберите репозиторий `wowsilizing-bot`
5.  Нажмите **Deploy Now**

5.2 Первая попытка развертывания

Railway автоматически начнет развертывание:

1.  Вы увидите экран с логами (текстом)
2.  Railway будет:
 - Скачивать код из GitHub
 - Устанавливать зависимости
 - Собирать Docker образ
 - Пытаться запустить бота
3.  **Первая попытка завершится ошибкой** — это нормально!
 - Причина: мы еще не добавили переменные окружения (токены и ключи)
4.  Не волнуйтесь, мы исправим это на следующем шаге!

Шаг 6: Настройка переменных окружения

 **Время: 5-7 минут**

Переменные окружения — это секретные настройки (токены, ключи), которые нужны боту для работы.

6.1 Открытие настроек

1.  В Railway, на странице вашего проекта, найдите вкладку **Variables**
2.  Нажмите на вкладку **Variables**

6.2 Добавление обязательных переменных

Добавьте эти переменные **обязательно**:

1 BOT_TOKEN

Имя: BOT_TOKEN
Значение: ваш_токен_от_BotFather

Как получить токен:

1. Откройте Telegram
2. Найдите бота [@BotFather](https://t.me/BotFather) (<https://t.me/BotFather>)
3. Отправьте команду /newbot (если создаете нового бота)
4. Или /token (если бот уже создан)
5. Скопируйте токен (выглядит как 1234567890:ABCdefGHIjklMN0pqrsTUVwxyz)

2 PREMIUM_USERNAME

Имя: PREMIUM_USERNAME
Значение: WowFUX

⚠ Важно: Пишите username **БЕЗ** символа @

6.3 Добавление опциональных переменных (для AI функций)

Если хотите использовать AI функции, добавьте эти ключи:

3 OPENAI_API_KEY (опционально)

Имя: OPENAI_API_KEY
Значение: sk-...ваш_ключ

Как получить:

1. Зарегистрируйтесь на <https://platform.openai.com/>
2. Перейдите в раздел **API Keys**: <https://platform.openai.com/api-keys>
3. Нажмите **Create new secret key**
4. Скопируйте ключ (начинается с sk-)

4 GOOGLE_API_KEY (опционально)

Имя: GOOGLE_API_KEY
Значение: ваш_ключ_gemini

Как получить:

1. Перейдите на <https://makersuite.google.com/app/apikey>
2. Нажмите **Create API Key**
3. Скопируйте ключ

5 ELEVENLABS_API_KEY (опционально)

Имя: ELEVENLABS_API_KEY
Значение: ваш_ключ_elevenlabs

Как получить:

1. Зарегистрируйтесь на <https://elevenlabs.io/>
2. Перейдите в **Profile Settings**
3. Скопируйте API Key

6 LOG_LEVEL (опционально)

Имя: LOG_LEVEL

Значение: INFO

Возможные значения: DEBUG , INFO , WARNING , ERROR

6.4 Как добавить переменную

Для каждой переменной:

1. Нажмите **New Variable** или **Add Variable**
2. В поле **Variable Name** введите имя (например: BOT_TOKEN)
3. В поле **Value** введите значение (ваш токен или ключ)
4. Нажмите **Add** или просто нажмите Enter
5. Повторите для всех переменных

6.5 Проверка переменных

После добавления вы должны увидеть список:

BOT_TOKEN = 1234567890:ABC... (скрыто)
 PREMIUM_USERNAME = WowFUX
 OPENAI_API_KEY = sk-... (скрыто)
 GOOGLE_API_KEY = ... (скрыто)
 ELEVENLABS_API_KEY = ... (скрыто)
 LOG_LEVEL = INFO



Безопасность: Railway автоматически скрывает значения переменных

Шаг 7: Запуск и тестирование

Время: 5-10 минут

Теперь запустим бота и проверим, что все работает!

7.1 Перезапуск развертывания

1. После добавления переменных Railway **автоматически перезапустит** бота
2. Если этого не произошло:
 - Перейдите на вкладку **Deployments**
 - Нажмите на последнее развертывание
 - Нажмите **Restart** или **Redeploy**

3. 📈 Следите за логами:

- Перейдите на вкладку **Logs**
- Вы увидите процесс запуска бота

7.2 Проверка успешного запуска

В логах вы должны увидеть примерно такие сообщения:

```
✓ Building...
✓ Installing dependencies...
✓ Starting bot...
✓ Bot started successfully
✓ Listening for updates...
```

Если видите ошибки — перейдите к разделу [Устранение неполадок](#)

7.3 Тестирование бота в Telegram

1. 📱 Откройте Telegram
2. 🔎 Найдите вашего бота по username (который вы указали в BotFather)
3. ⚡ Нажмите **Start** или отправьте команду `/start`
4. 🤖 Бот должен ответить приветственным сообщением!

7.4 Проверка основных функций

Попробуйте команды:

```
/start      - Приветствие и главное меню
/help       - Список всех команд
/stats     - Статистика использования
```

Попробуйте отправить боту:

- 📹 Видео (для обработки)
- 🔗 Ссылку на YouTube видео
- 🎵 Аудио файл

7.5 Проверка премиум функций

Если вы добавили AI ключи, проверьте премиум функции:

1. 🎙️ Отправьте голосовое сообщение (для транскрипции через Whisper)
2. 💬 Попробуйте AI чат (если реализован)
3. 🎵 Проверьте генерацию голоса (если реализована)

Устранение неполадок

Проблема: Бот не запускается

Симптомы: В логах Railway видите ошибки

Решения:**1. Проверьте переменные окружения:**

- Перейдите на вкладку **Variables**
- Убедитесь, что `BOT_TOKEN` и `PREMIUM_USERNAME` добавлены
- Проверьте, нет ли лишних пробелов в значениях

2. Проверьте токен бота:

- Откройте [@BotFather](https://t.me/BotFather) (<https://t.me/BotFather>) в Telegram
- Отправьте `/mybots`
- Выберите вашего бота
- Нажмите **API Token**
- Скопируйте токен и обновите в Railway

3. Проверьте логи:

- Вкладка **Logs** в Railway
- Найдите строку с ошибкой (обычно красным цветом)
- Прочтайте сообщение об ошибке

✖ Проблема: Бот не отвечает в Telegram

Симптомы: Бот запущен на Railway, но не отвечает на сообщения

Решения:**1. Проверьте статус бота:**

- В Railway на вкладке **Deployments** статус должен быть **Active** (зеленый)

2. Проверьте, что бот запущен в BotFather:

- Откройте [@BotFather](https://t.me/BotFather) (<https://t.me/BotFather>)
- Отправьте `/mybots`
- Выберите вашего бота
- Убедитесь, что бот не выключен

3. Перезапустите бота:

- В Railway перейдите на вкладку **Deployments**
- Нажмите **Restart**

4. Проверьте логи на ошибки:

- Вкладка **Logs**
- Ищите сообщения об ошибках

✖ Проблема: AI функции не работают

Симптомы: Бот работает, но AI функции (транскрипция, генерация голоса) не работают

Решения:**1. Проверьте API ключи:**

- Вкладка **Variables** в Railway
- Убедитесь, что добавлены `OPENAI_API_KEY`, `GOOGLE_API_KEY`, `ELEVENLABS_API_KEY`

2. Проверьте баланс на AI сервисах:

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

3. Проверьте права доступа:

- Убедитесь, что API ключи активны и не истекли

Проблема: Закончились кредиты на Railway

Симптомы: Railway остановил бота, сообщение о превышении лимита

Решения:

1. Проверьте использование:

- Перейдите в **Usage** на Railway
- Посмотрите, сколько кредитов использовано

2. Оптимизируйте использование:

- Уменьшите количество запросов к AI сервисам
- Проверьте, нет ли утечек памяти в логах

3. Добавьте карту (опционально):

- Railway попросит добавить карту для продолжения
- Вы можете установить лимит расходов
- Минимальный платеж: \$5

Проблема: Ошибка при загрузке файлов на GitHub

Симптомы: Не удается загрузить файлы через веб-интерфейс

Решения:

1. Проверьте размер файлов:

- GitHub ограничивает размер файла до 100 МБ
- Не загружайте большие PDF или видео файлы

2. Используйте метод создания файлов вручную:

- См. раздел [Метод 2: Создание файлов вручную](#)

3. Проверьте интернет соединение:

- Убедитесь, что интернет стабилен
- Попробуйте загрузить файлы по одному

Проблема: Railway не видит репозиторий

Симптомы: При создании проекта на Railway не видно вашего репозитория

Решения:

1. Обновите права доступа:

- Перейдите на <https://github.com/settings/installations>
- Найдите **Railway**
- Нажмите **Configure**
- Убедитесь, что репозиторий `wowsilizing-bot` выбран

2. Переподключите GitHub:

- В Railway перейдите в **Settings**
- Отключите GitHub
- Подключите заново

Нужна дополнительная помощь?

Если проблема не решена:

1.  Скопируйте сообщение об ошибке из логов Railway
 2.  Сделайте скриншот проблемы
 3.  Поиските ошибку в Google или ChatGPT
 4.  Обратитесь к разработчику бота
-

Поздравляем!

Если вы дошли до этого момента и бот работает — **вы молодец!** 

Теперь ваш бот:

-  Работает 24/7 на Railway
-  Код хранится на GitHub
-  Автоматически обновляется при изменениях
-  Полностью под вашим контролем

Что дальше?

-  Прочтите [QUICK_REFERENCE.md](#) (QUICK_REFERENCE.md) для быстрого доступа к важной информации
-  Изучите настройки бота в файле config.py
-  Следите за использованием кредитов на Railway
-  При необходимости обновляйте код на GitHub

Как обновить бота в будущем?

1. Внесите изменения в файлы на GitHub (через веб-интерфейс)
2. Railway автоматически обнаружит изменения
3. Бот перезапустится с новым кодом

Это все! Удачи с вашим ботом! 

Документация создана для WOwsilizing Bot

Версия: 1.0

Дата: Ноябрь 2024