



Пошаговое развертывание на Vercel



Подготовка проекта

1. Скачивание файлов

1. В ChatLLM Teams нажмите кнопку **“Files”** → скачайте ZIP
2. Распакуйте архив и перейдите в папку `workout_app/app`

2. Исправление конфигурации

Переименуйте файлы:

```
# В папке workout_app/app/  
mv next.config.js next.config.js.backup  
mv next.config.vercel.js next.config.js
```

3. Очистка кеша (важно!)

Удалите проблемные папки:

```
rm -rf .next  
rm -rf .build  
rm -rf node_modules
```



Создание GitHub репозитория

1. Создайте репозиторий

- Зайдите на [GitHub.com](https://github.com)
- Создайте **новый репозиторий** (например: `workout-tracker`)
- ☒ **НЕ добавляйте** README, .gitignore, license

2. Загрузите код

```
# В папке workout_app/app/  
git init  
git add .  
git commit -m "feat: workout tracker v1.0.1"  
git branch -M main  
git remote add origin https://github.com/ваш-username/workout-tracker.git  
git push -u origin main
```



Развертывание на Vercel

1. Подключение репозитория

1. Зайдите на vercel.com (<https://vercel.com>)
2. Нажмите **“New Project”**
3. Импортируйте ваш GitHub репозиторий

4. **Root Directory:** оставьте пустым (проект в корне)
5. **Framework Preset:** Next.js (должно определиться автоматически)

2. Настройки сборки

В разделе **Build and Output Settings:**

- ☒ **Build Command:** `yarn build` (по умолчанию)
- ☒ **Output Directory:** `.next` (по умолчанию)
- ☒ **Install Command:** `yarn install` (по умолчанию)

3. Переменные окружения

Добавьте в **Environment Variables:**

```
NODE_ENV=production
NEXTAUTH_URL=https://ваш-проект.vercel.app
NEXTAUTH_SECRET=ваш-секретный-ключ-минимум-32-символа
```

Для генерации NEXTAUTH_SECRET:

```
openssl rand -base64 32
```

4. Деплой

1. Нажмите **“Deploy”**
2. Ждите завершения сборки (2-3 минуты)



Решение частых ошибок

Ошибка: “Cannot find module”

Решение: Убедитесь что удалили `.next` и `node_modules`

Ошибка: TypeScript/ESLint ошибки

Решение: В `next.config.js` временно добавьте:

```
typescript: { ignoreBuildErrors: true },
eslint: { ignoreDuringBuilds: true },
```

Ошибка: “Prisma client not generated”

Решение: Добавьте в `package.json` scripts:

```
"postinstall": "prisma generate"
```

Ошибка: “Module not found: Can’t resolve ‘fs’”





Решение: Уже исправлено в `next.config.vercel.js`



Проверка работы

После деплоя проверьте:

1. **Главная страница** загружается ☒

2. Можно добавить тренировку 
3. Переключение между вкладками работает 
4. Сохранение данных в localStorage 
5. В Telegram (если запускать из бота) - работает Cloud Storage 





Настройка домена (опционально)

1. В Vercel Dashboard → Settings → Domains
2. Добавьте свой домен
3. Настройте DNS записи у регистратора
4. Обновите `NEXTAUTH_URL` на новый домен






Важные моменты

Для Telegram Web App:

-  **HTTPS обязателен** (Vercel предоставляет автоматически)
-  **CSP заголовки** уже настроены в `vercel.json`

Производительность:

-  **Кеширование** включено
-  **Оптимизация изображений** включена
-  **Минификация** включена



Использование

Как веб-приложение:

Просто откройте `https://ваш-проект.vercel.app`

Как Telegram Web App:

1. Создайте бота через @BotFather
2. Настройте Web App URL: `https://ваш-проект.vercel.app`
3. Добавьте команду `/start` с кнопкой "Открыть приложение"



Если что-то не работает

1. Проверьте логи в Vercel Dashboard → Functions → View Logs
2. Проверьте консоль браузера на ошибки JavaScript
3. Убедитесь что все environment variables настроены
4. Попробуйте повторный деплой: Git push → Vercel автоматически пересоберет

Готово! 🎉 Ваше приложение должно работать на Vercel.