

1. Введение

Flexberry Next — это мощный open-source фреймворк для создания современных веб- и гибридных мобильных приложений, построенный на базе фреймворка **Next.js**. Он предоставляет комплексный набор готовых клиентских компонентов, инструментов для работы с данными и интеграции с серверной частью, что значительно ускоряет разработку корпоративных приложений.

Цель данного документа — предоставить инструкцию по установке и первоначальной настройке программного обеспечения Flexberry Next, что необходимо для его развертывания и начала разработки.

2. Предварительные требования

Перед началом установки убедитесь, что на вашем компьютере установлено следующее программное обеспечение:

Компонент	Минимальная версия	Рекомендуемая версия	Примечание
Node.js (включая npm)	16.x	18.x LTS или новее	Необходим для работы Next.js и клиентских зависимостей.
Git	2.x	Последняя стабильная	Требуется для клонирования репозиториев.
СУБД (опционально)	—	PostgreSQL, MS SQL Server	Необходима для работы серверной части (Backend) сгенерированного приложения.
.NET SDK (опционально)	6.0	8.0 или новее	Требуется для сборки и запуска серверной части, сгенерированной Flexberry Designer.

3. Основные сценарии установки

Существует два основных пути для начала работы с Flexberry Next:

- **Сценарий 1 (Рекомендуемый): Использование Flexberry Designer.** Это основной промышленный подход, при котором вы создаете модель приложения в визуальном конструкторе, а затем автоматически генерируете полнофункциональный проект, включающий серверную часть на C# с Flexberry ORM и клиентскую часть на Flexberry Next.
- **Сценарий 2: Использование готового примера (GitHub).** Этот подход позволяет быстро ознакомиться с архитектурой и возможностями фреймворка, запустив готовый демонстрационный проект.

Далее каждый сценарий описан подробно.

4. Сценарий 1: Установка и запуск с помощью Flexberry Designer

4.1. Шаг 1. Установка Flexberry Designer

Flexberry Designer — это основной инструмент для проектирования и генерации приложений платформы Flexberry.

1. Перейдите на [официальный сайт Flexberry](#).
2. Зарегистрируйте учетную запись и авторизуйтесь.
3. В разделе "Flexberry Designer Desktop" скачайте установщик для вашей операционной системы (Windows).

Запустите скачанный установочный файл (например, setup.exe) и следуйте инструкциям мастера установки.

При первом запуске программы введите логин и пароль от сайта [Flexberry.net](#) для активации лицензии.

4.2. Шаг 2. Создание проекта и генерация приложения

1. В Flexberry Designer создайте новый проект, задав ему имя и идентификатор.
2. Спроектируйте модель предметной области: создайте классы, атрибуты, связи и формы.

Перейдите в настройки генерации проекта. Укажите, что клиентская часть должна быть сгенерирована с использованием **Flexberry Next (Next.js)**.

Настройте пути для генерации кода или укажите репозиторий GitHub, если требуется автоматическая публикация.

Запустите процесс генерации. В результате будет создана папка с проектом, содержащая:

- **Серверную часть (Backend):** Проект на C# (например, .sln файл), использующий Flexberry ORM для доступа к данным и предоставляющий RESTful/OData API.
- **Клиентскую часть (Frontend):** Проект на Next.js (Flexberry Next) со всеми сгенерированными формами, компонентами и маршрутами.

4.3. Шаг 3. Настройка и запуск сгенерированного приложения

4.3.1. Настройка и запуск серверной части (Backend)

1. Откройте сгенерированный .sln файл в среде разработки (например, Visual Studio или JetBrains Rider).
2. В файле конфигурации (например, appsettings.json) укажите строку подключения к вашей СУБД (PostgreSQL, MSSQL и т.д.).
3. Восстановите зависимости NuGet и выполните миграции базы данных (при необходимости).
4. Соберите и запустите проект. Серверная часть должна запуститься на локальном хосте (например, <https://localhost:5001>) и предоставлять API.

4.3.2. Настройка и запуск клиентской части (Frontend на Flexberry Next)

1. Откройте терминал и перейдите в папку сгенерированного клиентского приложения (например, Client).
2. Установите все необходимые зависимости npm:

bash

```
npm install
```

В файле конфигурации клиента (например, .env.local или в настройках config.js) укажите URL-адрес запущенного серверного API (Backend).

Запустите приложение в режиме разработки:

bash

```
npm run dev
```

5. После успешной сборки откройте браузер и перейдите по адресу <http://localhost:3000>. Вы должны увидеть сгенерированное приложение с формами и навигацией.

5. Сценарий 2: Установка и запуск готового примера (GitHub)

Этот подход подходит для быстрого ознакомления.

5.1. Шаг 1. Клонирование репозитория и установка зависимостей

1. Клонируйте репозиторий с примером приложения:

bash

```
git clone https://github.com/Flexberry/Flexberry.NextJs.Sample.git  
cd Flexberry.NextJs.Sample
```

Установите зависимости клиентской части:

bash

```
npm install
```

5.2. Шаг 2. Настройка и запуск серверной части

1. Для работы примера требуется работающий backend API. Вы можете:
 - Использовать демонстрационный API, если его адрес указан в конфигурации примера.
 - Запустить собственный сгенерированный backend (как описано в Сценарии 1).
2. Обновите конфигурационный файл в папке клиента, указав корректный URL API.

5.3. Шаг 3. Настройка и запуск клиентской части

1. Запустите клиентское приложение:

bash

```
npm run dev
```

2. Откройте <http://localhost:3000> в браузере для просмотра работающего примера.

6. Проверка успешной установки

Установка считается успешной, если:

1. Серверная часть (Backend) запускается без ошибок и предоставляет доступные эндпоинты API (например, <https://localhost:5001/odata>).
2. Клиентская часть (Frontend) на Flexberry Next запускается на <http://localhost:3000>.
3. В браузере отображается пользовательский интерфейс приложения (например, форма входа или список сущностей).
4. Клиентское приложение может выполнять запросы к API (например, загружать или сохранять данные), что свидетельствует о корректной связи между фронтеном и бэкендом.

7. Дополнительная конфигурация и возможности

После успешной установки вы можете настроить дополнительные возможности Flexberry Next:

- **Локализация:** изменение языка интерфейса через конфигурационные файлы.
- **Темы оформления:** кастомизация цветовой схемы и стилей компонентов.
- **Кастомизация компонентов:** использование механизма "скобок программиста" для безопасного расширения сгенерированного кода без риска потери изменений при перегенерации.
- **Интеграция с Docker:** для развертывания как клиентской, так и серверной части в контейнерах.

8. Устранение неполадок

Проблема	Возможная причина	Решение
Ошибка npm install	Проблемы с сетью или Node.js	Убедитесь, что Node.js установлен правильно. Попробуйте очистить кэш npm (npm cache clean --force) и повторить установку.
Клиент не может подключиться к backend	Неверный URL API в настройках клиента	Проверьте, что backend запущен, а адрес в конфигурации клиента (например, в .env.local) указан верно.
Ошибки при генерации в Flexberry Designer	Проблемы с лицензией или моделью	Убедитесь, что вы авторизованы в Designer, и проверьте корректность UML-модели
Ошибки C# компиляции	Отсутствующие пакеты NuGet или неверная строка подключения к БД	Восстановите пакеты NuGet в проекте и проверьте параметры подключения к базе данных в appsettings.json.

9. Заключение и полезные ссылки

Данный документ содержит полную инструкцию по установке и начальной настройке фреймворка Flexberry Next. Следуя указанным шагам, вы можете развернуть рабочее окружение для разработки современных веб-приложений.

Полезные ресурсы:

- Официальный сайт Flexberry Platform: <https://flexberry.net>
- Документация платформы: <https://flexberry.github.io>
- Репозиторий с примером Flexberry Next: <https://github.com/Flexberry/Flexberry.NextJs.Sample>
- Исходный код Flexberry ORM: <https://github.com/Flexberry/NewPlatform.Flexberry.ORM>

10. Контакты для получения технической поддержки

- По вопросам приобретения лицензий и технической помощи обращайтесь по адресу support@flexberry.net.