

АНАЛИЗ И СРАВНЕНИЕ СТЕКА

1 Обзор требований

Основная задача состоит в создании простого веб-интерфейса чата, напоминающего ChatGPT.

На базе React: допускается использование любой системы сборки (Vite, CRA, Next.js), что предоставляет гибкость в выборе фреймворка.

Форма ввода сообщений: функциональная область для ввода пользовательских сообщений.

Отображение истории чатов: визуальное представление прошлых бесед слева.

Навигация по чатам: возможность переключаться между различными сессиями чата.

Отображение истории сообщений: показ сообщений внутри выбранного чата.

Базовая валидация: например, запрет отправки пустых строк.

Чистый и понятный UI: акцент на пользовательский опыт и читаемость кода.

Дополнительные возможности:

Интеграция с БЯМ: подключение к большой языковой модели (например, через N8N, Flowise, OpenAI API).

Поддержка Markdown: отображение форматированных ответов.

2 Идея приложения и стек

Приложение для чата с персонажами, где пользователь сначала вводит имя персонажа (например, Илон Маск), а затем бот отвечает в стиле этого персонажа.

Фронтенд-фреймворк Next.js, стилизация TailwindCSS, локальное хранилище IndexedDB, обертка для работы с IndexedDB DexieJS, уникальные идентификаторы uuidv4, рендеринг Markdown react-markdown, интеграция с LLM OpenRouter

3 Анализ архитектуры приложения

Выбор – Feature-Sliced Design. FSD является современной архитектурной методологией, специально созданной для построения масштабируемых фронтенд-приложений.

Альтернативы:

Atomic Design: фокусируется на UI-компонентах, но слабее в бизнес-логике

Domain-Driven Design (DDD): более сложный для небольших проектов

Простая папочная структура: быстрее в реализации, но хуже масштабируется

FSD упрощает поддержку и масштабирование.

4 Анализ фреймворка

Next.js предоставляет множество возможностей из коробки: SSR, статическую генерацию, встроенную маршрутизацию и оптимизацию. Встроенная маршрутизация App Router идеально подходит для навигации между чатами.

Альтернативы:

Vite + React Router: быстрее в разработке, легче настройка

Create React App: проще, но устарел

Remix: отличная работа с данными, но требуется время на изучение

Next.js позволяет держать всё в одном месте — UI, API, маршруты, SSR — и отлично подходит для масштабируемого chat-приложения.

5 Анализ стилизации

TailwindCSS – утилитарный подход, применение стилей непосредственно в JSX. Быстрая разработка и стилизация, особенно для пользовательских дизайнов.

Альтернативы

CSS Modules: больше контроля, но медленнее разработка

Styled Components: устарел, хорошая типизация, но bundle размер

Chakra UI/MUI: готовые компоненты, но менее кастомизируемые

Выбор Tailwind оправдан, т.к. он даёт минимальный и, самое главное, быстрый CSS без потери гибкости.

6 Анализ хранения данных

IndexedDB – полноценная база данных в браузере. Предназначена для хранения больших объемов структурированных данных (>50 МБ, зависит от браузера, ограничено дисковым пространством). Асинхронный API, не блокирует основной поток. Может хранить сложные типы данных (объекты, массивы, бинарные данные). Поддерживает транзакции для согласованности данных. Поддерживает индексы для эффективных запросов. Обеспечивает возможности автономной работы.

Dexie.js – обертка API IndexedDB, упрощает запросы и обеспечивает реактивность (хук `useLiveQuery`).

Альтернативы:

LocalStorage: проще, но ограничения по размеру и типам данных

Серверная БД: даёт постоянное хранение, доступ с разных устройств, централизованную логику, но нужен собственный бэкенд/авторизация/базы данных

Для окончательного продукта лучше предусмотреть синхронизацию с бэкендом, так как данные живут только в браузере пользователя, не синхронизируются с сервером и могут потеряться при очистке. Но для простого приложения, но в данном случае для упрощения IndexedDB с Dexie.js являются хорошим вариантом.

7 Анализ AI-интеграции

OpenRouter принимает более гибкий подход, что делает его привлекательным вариантом для обычных пользователей, даёт доступ к множеству моделей через единый API и возможность бесплатного доступа к моделям.

Альтернативы:

OpenAI API: дороже, нужен VPN

Local LLM: бесплатно, но требует развертывания и больших усилий для применения

OpenRouter даёт бесплатный доступ к моделям и легко подключается.

8 Анализ дополнительных библиотек

react-markdown: лучшее решение для рендеринга форматированных ответов (код, списки) с защитой от XSS-атак.

uuidv4: гарантирует уникальность ID чатов, исключая коллизии.

9 Выводы

FSD + Next.js даёт архитектурную чистоту и готовность к масштабированию.

OpenRouter + react-markdown решает ключевую задачу (AI-чат) с минимальными затратами.

DexieJS обеспечивает надёжное хранение данных без бэкенда.

10 Источники

Для анализа использовались модели chat gpt-4o deep research, gemini 2.5 flash deep research, claude sonnet 4 explanatory web search, deepseek deepthink r1 search.