

УДК 004

ВЫБОР ИНСТРУМЕНТОВ ДЛЯ РЕАЛИЗАЦИИ ПЛАТФОРМЫ РАСПРОСТРАНЕНИЯ МУЗЫКАЛЬНЫХ ПРОИЗВЕДЕНИЙ

Драпкин Н.А., студент гр. ПИб-201,

Василевский В.А., студент гр. ПИб-201,

Научный руководитель: Верёвкин С.А., старший преподаватель

Кузбасский государственный технический университет

имени Т. Ф. Горбачева. Кемерово

Аннотация: В статье рассматривается важный этап разработки платформы для распространения музыкальных произведений - выбор необходимых инструментов и технологий. Проводим обзор основных аспектов выбора языков программирования, фреймворков, а также наборов технологий для бэкенда. Особое внимание уделяется сравнительному анализу различных инструментов и технологий, а также их применимости к задачам создания платформы для распространения музыкальных произведений.

Ключевые слова: программирование, анализ, начинающие разработчики, IT-индустрия, технологии для разработки

Введение

В наше время ИТ-технологии внесли революционные изменения в музыкальную индустрию, дав возможность музыкантам и продюсерам распространять свои музыкальные произведения. Разработка платформы для удобного распространения музыкальных произведений становится наиболее актуальной и востребованной. Для подобного рода задач выбор подходящих инструментов и технологий для реализации таких платформ играет решающую роль. Цель статьи заключается в рассмотрении ключевых аспектов выбора инструментов и технологий при разработке платформы для распространения музыкальных произведений.

Сравнение технологий для Frontend

Выбор языка программирования. TypeScript. TypeScript является надстройкой над языком JavaScript, который используется в браузерном движке V8. TypeScript обеспечивает строгую проверку типов на этапе компиляции, что позволяет обнаруживать потенциальные ошибки, что упрощает разработку веб-приложений. Для проекта платформы распространения музыкальных произведений, где важны безопасность и надежность, использование TypeScript является предпочтительным выбором.

Выбор фреймворка. VueJS. Проводя сравнительный анализ среди современных лидирующих фреймворков ReactJS[1], VueJS[2] и Angular[3] предпочтительным выбором стал VueJS на основе ряда причин:

- Имеет собственный маршрутизатор, разработанный и поддерживаемый официальным разработчиком.
- Возможность писать HTML5, CSS, JS в одном файле (Single-File Components).
- Двустороннее связывание – возможность менять контент внутри DOM-элемента одновременно с изменением переменной с использованием всего одной директивы. Данная возможность повышает удобство и скорость разработки.

Сравнение технологий для Backend

Выбор языка программирования. C#. Это язык программирования общего назначения, разработанный Microsoft, широко используемый для создания веб-приложений и служб. C# обладает строгой типизацией, современным синтаксисом и богатым набором библиотек и инструментов разработки. Он также интегрируется с платформой .NET, что обеспечивает высокую производительность и безопасность приложений.

Выбор хранилищ данных. При создании платформы распространения музыкальных произведений нужно учитывать виды сохраняемых данных. Соответственно, для каждого формата данных рекомендуется использовать предназначеннное хранилище:

- MINIO: Это распределенное хранилище объектов, совместимое с Amazon S3, предназначеннное для хранения и управления большими объемами данных. MINIO обеспечивает высокую доступность, масштабируемость и надежность данных, что делает его подходящим выбором для хранения музыкальных файлов и метаданных.
- PsSQL (PostgreSQL): Это мощная реляционная база данных с открытым исходным кодом, широко используемая для разработки веб-приложений. PsSQL обеспечивает высокую производительность, надежность и расширяемость, а также поддерживает множество расширений и инструментов для работы с данными.
- REDIS: Это высокопроизводительная система управления данными, используемая для кэширования, хранения сессий и обработки сообщений. REDIS обеспечивает быстрый доступ к данным и поддерживает различные структуры данных, такие как строки, хэши, списки и множества, что делает его идеальным выбором для обработки данных в реальном времени на платформе распространения музыкальных произведений.

Выбор веб-сервера. Для стабильной и бесперебойной работы платформы и взаимодействия с ней необходимо программное обеспечение, которое

принимает HTTP-запросы от клиентов и выдается им HTTP-ответы. Таким ПО выступают веб-серверы:

- NGINX[4]: Это высокопроизводительный веб-сервер и прокси-сервер, который обеспечивает быструю и надежную доставку контента в сети. NGINX обладает масштабируемостью и эффективностью при обработке больших объемов запросов, а также поддерживает ряд расширений и интеграций. Также предоставляется способ распределения запросов к серверу в кластере.
- Apache[5]: Это один из самых распространенных веб-серверов в мире, широко используемый для развертывания веб-приложений и служб. Apache обеспечивает стабильность, надежность и гибкость в настройке, а также поддерживает множество модулей и расширений для работы с веб-сайтами и приложениями.

Для разработки платформы в нашем случае предпочтительно использовать NGINX, который имеет возможность балансировки нагрузки на сервер.

Заключение

В данной статье был рассмотрен важный этап разработки платформы для распространения музыкальных произведений - выбор инструментов и технологий. Мы провели анализ различных аспектов выбора программ для дизайна, языков программирования, фреймворков, баз данных, прокси-серверов, систем контроля версий и контейнеризации. Выбор подходящих инструментов является критическим для успешной реализации проекта и может оказать значительное влияние на производительность, масштабируемость и надежность создаваемой платформы.

Исходя из проведенного анализа, можно сделать вывод о том, что каждый инструмент имеет свои преимущества и недостатки, и выбор должен быть основан на конкретных потребностях и требованиях проекта, а также на опыте и предпочтениях команды разработчиков.

Список литературы.

1. Документация React [Электронный ресурс] – URL:
<https://reactjs.org/docs/getting-started.html> / React (reactjs.org) (дата обращения: 21.03.2024)
2. Документация VueJS [Электронный ресурс] – URL:
<https://vuejs.org/v2/guide/> / Vue (vuejs.org) (дата обращения: 21.03.2024)
3. Документация Angular [Электронный ресурс] – URL:
<https://angular.io/docs> / Angular (angular.io) (дата обращения: 21.03.2024)
4. Документация NGINX [Электронный ресурс] – URL:
<https://nginx.org/en/docs/> / NGINX (nginx.org) (дата обращения: 21.03.2024)
5. Документация Apache [Электронный ресурс] – URL:
<https://httpd.apache.org/docs/> / Apache (apache.org) (дата обращения: 21.03.2024)

Драпкин Никита Андреевич, студент направления «Прикладная информатика», ПИб-201, 4 курс, drapkin2013@yandex.ru, Россия, Кемерово, ФГБОУ ВО «Кузбасский государственный технический университет имени Т.Ф. Горбачёва».

Василевский Валерий Александрович, студент направления «Прикладная информатика», ПИб-201, 4 курс, valera200208@gmail.com, Россия, Кемерово, ФГБОУ ВО «Кузбасский государственный технический университет имени Т.Ф. Горбачёва».