Оглавление

[ВВЕДЕНИЕ 2](#__RefHeading___Toc1685_251362023)

[1. Теоритическая часть 5](#__RefHeading___Toc1687_251362023)

[1.1 Анализ предметной области 5](#__RefHeading___Toc1260_2890793370)

[1.2 Анализ существующих решений 6](#__RefHeading___Toc1262_2890793370)

[1.3 Требования к разрабатываемому решению 8](#__RefHeading___Toc1264_2890793370)

[1.4 Обоснование необходимости разработки 9](#__RefHeading___Toc1266_2890793370)

[2. Технологическая часть 11](#__RefHeading___Toc1689_251362023)

[2.1 Анализ и выбор JavaScript-фреймворков 11](#__RefHeading___Toc1268_2890793370)

[2.2 Сравнение инструментов управления состоянием 12](#__RefHeading___Toc1270_2890793370)

[2.3 Анализ и выбор инструментов стилизации 13](#__RefHeading___Toc1272_2890793370)

[2.4 Выбор дополнительных инструментов 15](#__RefHeading___Toc1274_2890793370)

[2.5 Обоснование технологического стека 16](#__RefHeading___Toc1276_2890793370)

[2.6 Разработка проекта решений 16](#__RefHeading___Toc1278_2890793370)

[3 ПРАКТИЧЕСКАЯ ЧАСТЬ 19](#__RefHeading___Toc1691_251362023)

[3.1 Разработка аутентификации 20](#__RefHeading___Toc1693_251362023)

[3.2 Разработка общих компонентов 22](#__RefHeading___Toc1695_251362023)

[3.3 Разработка ключевых страниц сайта. 24](#__RefHeading___Toc1697_251362023)

[3.4 Разработка функционала прослушивания. 30](#__RefHeading___Toc1699_251362023)

[3.5 Инструкция по эксплуатации 35](#__RefHeading___Toc1701_251362023)

[ЗАКЛЮЧЕНИЕ 37](#__RefHeading___Toc1703_251362023)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 38](#__RefHeading___Toc1705_251362023)

# ВВЕДЕНИЕ

В современном мире цифровизация проникает во все сферы жизни, включая творческие и культурные области. Одной из таких областей является реставрация аудиозаписей, которая позволяет сохранять и восстанавливать исторически значимые материалы, такие как речи, музыка или архивные записи. Восстановление старых аудиофайлов — это не только технический процесс, но и способ сохранить культурное наследие, вернуть к жизни голоса прошлого. Однако ручная реставрация аудио требует значительных временных и профессиональных ресурсов, что делает процесс трудоемким и дорогостоящим. Автоматизация этого процесса с помощью специализированного программного обеспечения может значительно упростить задачу, повысить качество результата и сделать реставрацию доступной для широкого круга пользователей.

Разработка фронтенд-составляющей сервиса для реставрации аудио «Голос победы» направлена на создание удобного и интуитивно понятного интерфейса, который позволит пользователям, не обладающим глубокими техническими знаниями, загружать аудиофайлы, управлять процессом реставрации и получать качественные результаты. Проект ориентирован на обработку исторических аудиозаписей, в частности связанных с Великой Отечественной войной, что придает ему особую социальную значимость. Удобный интерфейс и продуманный пользовательский опыт играют ключевую роль в популяризации таких сервисов, делая их доступными для музеев, архивов, исследователей и частных пользователей.

**Цель курсовой работы** заключается в разработке фронтенд-составляющей веб-приложения «Голос победы», обеспечивающей удобное взаимодействие пользователя с сервисом реставрации аудиозаписей.

**Задачи работы** включают следующие этапы:

1. Проведение анализа предметной области реставрации аудиозаписей.

2. Изучение существующих решений для обработки аудио и их интерфейсов.

3. Формирование требований к фронтенд-составляющей разрабатываемого сервиса.

4. Анализ и выбор технологического стека для реализации интерфейса.

5. Разработка фронтенд-части приложения в соответствии с требованиями.

6. Проведение тестирования пользовательского интерфейса.

7. Подготовка отчета о проделанной работе.

8. Представление результатов и защита курсовой работы.

**Объектом курсовой работы** является процесс автоматизации реставрации аудиозаписей с использованием веб-технологий.

**Предметом курсовой работы** выступает разработка фронтенд-составляющей сервиса «Голос победы», включающая проектирование и реализацию пользовательского интерфейса.

**Методы, применяемые в работе**, включают анализ, синтез, сравнение и моделирование. Анализ используется для изучения предметной области и существующих аналогов, синтез — для формирования требований и проектирования интерфейса, сравнение — для выбора подходящих технологий, а моделирование — для создания прототипов и тестирования функциональности. Практическая реализация основана на современных подходах к разработке веб-приложений, включая компонентный подход и использование популярных фреймворков.

**Информационная база курсовой работы** включает открытые источники, доступные в сети Интернет, такие как документация по фронтенд-фреймворкам, статьи по разработке пользовательских интерфейсов, а также материалы курса «Создание программного обеспечения», предоставленные через систему дистанционного обучения РТУ МИРЭА.

В итоге, в отчете будет представлен процесс разработки фронтенд-составляющей сервиса «Голос победы», включая теоретический анализ предметной области, проектирование интерфейса, выбор технологий, реализацию и тестирование. Особое внимание уделено созданию интуитивно понятного интерфейса, который упрощает взаимодействие с сервисом и делает процесс реставрации аудио доступным для пользователей с разным уровнем подготовки.

# Теоритическая часть

## 1.1 Анализ предметной области

Реставрация аудиозаписей представляет собой процесс восстановления качества звука старых или поврежденных аудиофайлов, таких как записи речей, музыкальных произведений или исторических архивов. Основной целью является устранение шумов, щелчков, искажений и других дефектов, а также повышение разборчивости звука. В рамках сервиса «Голос победы» акцент сделан на обработке аудиозаписей, связанных с Великой Отечественной войной, включая речи ветеранов, радиопередачи и архивные записи военных событий. Эти материалы обладают высокой исторической и культурной ценностью, но их качество часто ухудшено из-за устаревших технологий записи, физического износа носителей или неправильного хранения.

Основные информационные процессы, реализуемые в реставрации аудио, включают следующие этапы:

1. Загрузка аудиофайлов в систему для последующей обработки.

2. Извлечение метаданных из аудиофайла для идентификации и классификации записи.

3. Применение ИИ-пайплайна для автоматической обработки аудио, включая устранение шумов и улучшение качества звука.

4. Визуализация ожидания процесса обработки для информирования пользователя о прогрессе.

5. Редактирование метаданных записи для уточнения информации о файле.

6. Сохранение обработанной записи, публикация в общей библиотеке или добавление в личную коллекцию пользователя.

7. Исследование общей библиотеки аудиозаписей, предоставляющей доступ к историческим материалам.

8. Использование личной коллекции аудиофайлов пользователя для удобного управления записями.

9. Прослушивание аудиозаписей через встроенный плеер.

Ключевые задачи фронтенд-составляющей заключаются в создании удобного интерфейса, обеспечивающего загрузку файлов, отображение статуса обработки, редактирование метаданных, доступ к библиотеке и личной коллекции, а также прослушивание записей. Ручная реставрация аудио сопряжена с рядом сложностей: необходимостью глубоких знаний в области звукообработки, использованием профессионального ПО (например, Adobe Audition или Audacity) и значительными временными затратами. Это ограничивает доступ к процессу для широкой аудитории, включая музеи, архивы и частных пользователей без специальных навыков. Автоматизированные веб-сервисы, такие как «Голос победы», решают эти проблемы, предоставляя интуитивный интерфейс и автоматизированные алгоритмы обработки.

В итоге, автоматизация реставрации аудио позволяет:

- Сократить время, необходимое для обработки записей.

- Сделать реставрацию доступной для непрофессионалов.

- Сохранить исторически значимые аудиоматериалы.

- Минимизировать ошибки, связанные с ручной обработкой.

## **1.2 Анализ существующих решений**

Для формирования концепции сервиса «Голос победы» был проведен анализ существующих программных продуктов для реставрации аудио. Рассмотрены следующие решения: iZotope RX, Adobe Audition, Audacity и Audio Restoration Online.

**iZotope RX** — профессиональное ПО, популярное среди звукорежиссеров. Оно предоставляет инструменты для удаления шумов, восстановления искаженных участков и улучшения качества звука. Преимущества включают высокую точность обработки и широкий функционал, такой как спектральный анализ и подавление кликов. Однако сложный интерфейс затрудняет использование новичками, а высокая стоимость делает программу недоступной для частных пользователей или малого бизнеса. Кроме того, iZotope RX является десктопным приложением, что исключает его применение в веб-среде.

**Adobe Audition** — профессиональный инструмент для обработки аудио, интегрированный в экосистему Adobe. Программа поддерживает функции реставрации, включая удаление шумов, восстановление частот и нормализацию громкости. Среди плюсов — высокая производительность и возможность интеграции с другими продуктами Adobe. Однако, как и iZotope RX, Adobe Audition требует значительных навыков и не предназначена для веб-доступа, что ограничивает ее использование в автоматизированных сервисах.

**Audacity** — бесплатное ПО с открытым исходным кодом, ориентированное на любителей. Оно поддерживает базовые функции реставрации, такие как удаление шумов и эквализация. Преимущества включают доступность и простоту освоения, но функциональность ограничена, а интерфейс выглядит устаревшим. Audacity не поддерживает веб-интеграцию, что делает его неподходящим для онлайн-платформ.

**Audio Restoration Online** — веб-сервис, предлагающий базовые возможности реставрации аудио. Пользователи могут загружать файлы, автоматически устранять шумы и экспортировать результаты. Преимущества заключаются в простоте использования и доступности через браузер, но качество обработки уступает профессиональным инструментам, а интерфейс часто перегружен рекламой, что ухудшает пользовательский опыт.

Для наглядности проведено сравнение решений по ключевым критериям: простота интерфейса, функциональность, доступность в веб-среде и наличие функций музыкального сервиса.

*Таблица 1.1 — Сравнение существующих решений для реставрации аудио*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерий | iZotope RX | Adobe Audition | Audacity | Audio Restoration Online |
| Простота интерфейса | - | - | + | + |
| Функциональность | + | + | - | - |
| Доступность в веб-среде | - | - | - | + |
| Наличие функций музыкального сервиса | - | - | - | - |

Анализ показывает, что профессиональные решения (iZotope RX, Adobe Audition) обладают высокой функциональностью, но сложны в использовании и не поддерживают веб-доступ. Audacity прост в освоении, но ограничен по возможностям. Веб-сервисы, такие как Audio Restoration Online, удобны, но уступают в качестве обработки. При этом ни одно из решений не предлагает функционал музыкального сервиса, включающего библиотеки записей, личные коллекции и встроенный плеер. Это подчеркивает актуальность разработки сервиса «Голос победы», сочетающего реставрацию аудио с элементами музыкальной платформы.

## 1.3 Требования к разрабатываемому решению

На основе анализа предметной области и существующих решений сформированы требования к фронтенд-составляющей сервиса «Голос победы». Они направлены на обеспечение удобства, функциональности и доступности для пользователей, с учетом специфики информационных процессов.

*Таблица 1.2 — Требования к фронтенд-составляющей*

|  |  |  |
| --- | --- | --- |
| **№** | **Требование** | **Описание** |
| 1 | Регистрация и аутентификация | Реализация регистрации и входа с помощью сторонних OAuth провайдеров для доступа к личной коллекции. |
| 2 | Загрузка аудиофайлов | Интерфейс для загрузки файлов формата MP3. |
| 3 | Визуализация ожидания | Отображение страницы ожидания обработки с индикатором прогресса. |
| 4 | Редактирование метаданных | Интерфейс для изменения метаданных аудиозаписи (название, автор и т.д.). |
| 5 | Экспорт и управление записями | Возможность скачивания обработанных файлов, публикации в общей библиотеке или добавления в личную коллекцию. |
| 6 | Доступ к библиотеке | Интерфейс для просмотра и поиска аудиозаписей в общей библиотеке. |
| 7 | Управление личной коллекцией | Функционал для просмотра, записей из личной коллекции. |
| 8 | Прослушивание записей | Встроенный аудиоплеер для воспроизведения записей из библиотеки или коллекции. |
| 9 | Интуитивный интерфейс | Простой и эргономичный дизайн. |
| 10 | Адаптивность | Поддержка корректного отображения и работы на устройствах с разными разрешениями экрана. |

Эти требования обеспечивают создание функционального интерфейса, который упрощает процесс реставрации аудио, предоставляет доступ к библиотеке записей и личной коллекции, а также поддерживает удобное прослушивание. Особое внимание уделено интуитивности и адаптивности, чтобы сервис был доступен для пользователей с разным уровнем подготовки, включая непрофессионалов.

## 1.4 Обоснование необходимости разработки

Разработка сервиса «Голос победы» обусловлена рядом факторов, выявленных в ходе анализа. Во-первых, существующие решения для реставрации аудио либо ориентированы на профессионалов и требуют специальных навыков (iZotope RX, Adobe Audition), либо предоставляют ограниченный функционал и низкое качество обработки (Audio Restoration Online). Во-вторых, ни один из рассмотренных продуктов не предлагает интеграцию функций музыкального сервиса, таких как общая библиотека записей, личные коллекции и встроенный плеер, что особенно важно для работы с историческими материалами. В-третьих, автоматизация реставрации с использованием ИИ-пайплайна позволяет снизить временные затраты и сделать процесс доступным для широкой аудитории, включая музеи, архивы и частных пользователей.

Сервис «Голос победы» решает проблему сохранения культурного наследия, предоставляя инструмент для восстановления и популяризации аудиозаписей Великой Отечественной войны. Удобный интерфейс и автоматизированная обработка устраняют барьеры, связанные с ручной реставрацией, а функции библиотеки и личной коллекции делают сервис уникальным в сравнении с аналогами. Это позволяет позиционировать проект как социально значимый, способствующий сохранению исторической памяти.

В итоге, теоретическая часть определила ключевые аспекты предметной области, выявила недостатки существующих решений и обосновала необходимость разработки нового веб-сервиса. Сформированные требования создают основу для дальнейшего проектирования и реализации фронтенд-составляющей, обеспечивая удобство, функциональность и доступность для пользователей.

# Технологическая часть

## ****2.1 Анализ и выбор JavaScript-фреймворков****

Для реализации фронтенд-составляющей сервиса «Голос победы» был проведен сравнительный анализ популярных JavaScript-фреймворков: React, Vue.js и Angular. Выбор фреймворка определялся такими критериями, как простота синтаксиса, функциональность, гибкость, производительность и поддержка сообщества. Эти критерии учитывают потребности проекта в создании интуитивного и адаптивного интерфейса с минимальными затратами времени на разработку.

**React** — библиотека для построения пользовательских интерфейсов, использующая компонентный подход и виртуальный DOM для повышения производительности. Преимущества включают гибкость, обширную экосистему библиотек и широкую поддержку сообщества. React хорошо подходит для динамических приложений, таких как «Голос победы», где требуется отображение прогресса обработки и взаимодействие с библиотекой записей. Однако необходимость настройки дополнительных инструментов, таких как роутинг или управление состоянием, увеличивает сложность и время разработки.

**Vue.js** — прогрессивный фреймворк, сочетающий простоту и функциональность. Его интуитивный синтаксис и встроенная реактивность упрощают создание компонентов и управление состоянием. Vue.js требует минимальной конфигурации, что делает его подходящим для быстрой разработки MVP. Гибкость фреймворка позволяет легко интегрировать сторонние библиотеки, а поддержка сообщества обеспечивает доступ к современным инструментам. Недостатки проявляются при масштабировании крупных проектов, но для задач сервиса «Голос победы» это не критично.

**Angular** — полноценный фреймворк с жесткой архитектурой, предоставляющий встроенные инструменты для роутинга, форм и HTTP-запросов. Он подходит для сложных корпоративных приложений благодаря строгой типизации и модульной структуре. Однако высокий порог входа, сложный синтаксис и избыточная функциональность делают Angular менее подходящим для небольшого проекта, где важны скорость и простота.

Для наглядности проведено сравнение фреймворков по ключевым критериям.

Таблица 2.1 — Сравнение JavaScript-фреймворков

| Критерий | React | Vue.js | Angular |
| --- | --- | --- | --- |
| Простота синтаксиса | - | + | - |
| Функциональность | + | + | + |
| Гибкость | + | + | - |
| Производительность | + | + | - |
| Поддержка сообщества | + | + | + |

С учетом анализа Vue.js выбран как оптимальный фреймворк для разработки фронтенд-составляющей сервиса «Голос победы». Его простота, гибкость и достаточная функциональность обеспечивают быструю реализацию интерфейса, соответствующего требованиям проекта.

## ****2.2 Сравнение инструментов управления состоянием****

Для управления глобальным состоянием приложения в рамках Vue.js были рассмотрены две библиотеки: Pinia и Vuex. Выбор инструмента определялся простотой интеграции, лаконичностью кода, поддержкой современных подходов и совместимостью с Vue.js.

**Pinia** — современная библиотека управления состоянием, разработанная специально для Vue 3. Она предоставляет интуитивный API, основанный на композиционном подходе, что упрощает работу с данными, такими как список аудиозаписей, метаданные или статус обработки. Pinia поддерживает прямую модификацию состояния, минимизируя шаблонный код. Модульная структура позволяет разделять хранилище на независимые модули, что удобно для управления различными аспектами приложения.

**Vuex** — более ранняя библиотека, долгое время являвшаяся стандартом для Vue.js. Она использует централизованное хранилище с четким разделением на мутации, действия и геттеры. Vuex обеспечивает стабильность и хорошо документирована, но ее структура требует написания большего объема кода, особенно для простых операций. Кроме того, Vuex менее оптимизирована для Vue 3, что снижает ее привлекательность для новых проектов.

Сравнение библиотек представлено в таблице.

Таблица 2.2 — Сравнение инструментов управления состоянием

| Критерий | Pinia | Vuex |
| --- | --- | --- |
| Простота API | + | - |
| Лаконичность кода | + | - |
| Интеграция с Vue 3 | + | - |
| Поддержка TypeScript | + | - |
| Модульность | + | + |

Pinia выбрана для реализации сервиса «Голос победы» благодаря простоте, поддержке современных стандартов и эффективной работе с Vue 3. Эта библиотека обеспечивает удобное управление состоянием, минимизируя сложность разработки.

## ****2.3 Анализ и выбор инструментов стилизации****

Для стилизации интерфейса были рассмотрены CSS-фреймворки и подходы: Tailwind CSS, Bootstrap и пользовательский CSS. Критерии выбора включали скорость разработки, адаптивность, объем кода и удобство поддержки.

**Tailwind CSS** — утилитарный фреймворк, предоставляющий готовые классы для стилизации элементов. Он позволяет быстро создавать адаптивные интерфейсы, минимизируя написание пользовательских стилей. Tailwind CSS интегрируется непосредственно в шаблоны компонентов, что упрощает поддержку и уменьшает объем CSS-файлов. Его гибкость обеспечивает точную настройку дизайна, а встроенные адаптивные классы гарантируют корректное отображение на разных устройствах.

**Bootstrap** — популярный CSS-фреймворк с готовыми компонентами, такими как кнопки, формы и навигационные панели. Он ускоряет разработку за счет унифицированного дизайна, но его компоненты имеют фиксированный стиль, что ограничивает кастомизацию. Bootstrap также увеличивает объем кода из-за зависимости от JavaScript для интерактивных элементов, что нежелательно для легковесного приложения.

**Пользовательский CSS** — подход, предполагающий написание стилей с нуля. Он предоставляет максимальную свободу в дизайне, но требует значительных временных затрат и усложняет поддержку, особенно при масштабировании проекта. Отсутствие готовых решений для адаптивности делает этот вариант менее предпочтительным.

Сравнение инструментов стилизации представлено в таблице.

Таблица 2.3 — Сравнение инструментов стилизации

| Критерий | Tailwind CSS | Bootstrap | Пользовательский CSS |
| --- | --- | --- | --- |
| Скорость разработки | + | + | - |
| Адаптивность | + | + | - |
| Объем кода | + | - | - |
| Удобство поддержки | + | - | - |
| Гибкость дизайна | + | - | + |

Tailwind CSS выбран для стилизации интерфейса сервиса «Голос победы» благодаря скорости разработки, адаптивности и минимальному объему кода. Этот фреймворк обеспечивает создание эргономичного и современного дизайна, соответствующего требованиям проекта.

## ****2.4 Выбор дополнительных инструментов****

Для реализации функционала извлечения метаданных из аудиофайлов была выбрана библиотека **jsmediatags**. Она обеспечивает асинхронную обработку MP3-файлов, позволяя извлекать данные, такие как название, автор, необходимые для редактирования метаданных. Библиотека легковесна, проста в интеграции и совместима с JavaScript, что делает ее подходящей для задач сервиса. Jsmediatags поддерживает работу с основным форматом проекта (MP3) и не требует сложной настройки, что ускоряет разработку.

Для взаимодействия с бэкендом выбрана библиотека **Axios**, обеспечивающая удобную работу с HTTP-запросами. Axios поддерживает асинхронные операции, обработку ошибок и настройку заголовков, что необходимо для загрузки файлов, аутентификации и получения данных. Ее простота и широкая поддержка сообщества делают библиотеку оптимальной для интеграции фронтенда с API.

Для администрирования данных в проекте использовалась платформа Hasura, которая предоставляет GraphQL API для управления записями, метаданными и пользовательскими данными. Hasura упрощает доступ к базе данных, обеспечивая гибкость в настройке запросов и высокую производительность, что поддерживает административные функции сервиса.

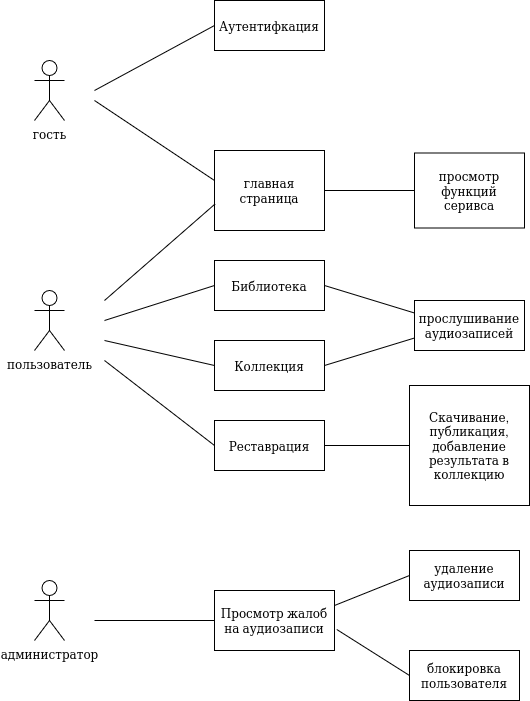
## ****2.5 Обоснование технологического стека****

Выбранный технологический стек включает Vue.js, Pinia, Tailwind CSS, jsmediatags и Axios. Vue.js обеспечивает гибкую и быструю разработку компонентного интерфейса, Pinia упрощает управление состоянием, а Tailwind CSS ускоряет создание адаптивного дизайна. Jsmediatags решает задачу извлечения метаданных, а Axios гарантирует надежное взаимодействие с бэкендом. Этот стек соответствует требованиям проекта по функциональности, производительности и удобству, а также позволяет минимизировать затраты времени на разработку.

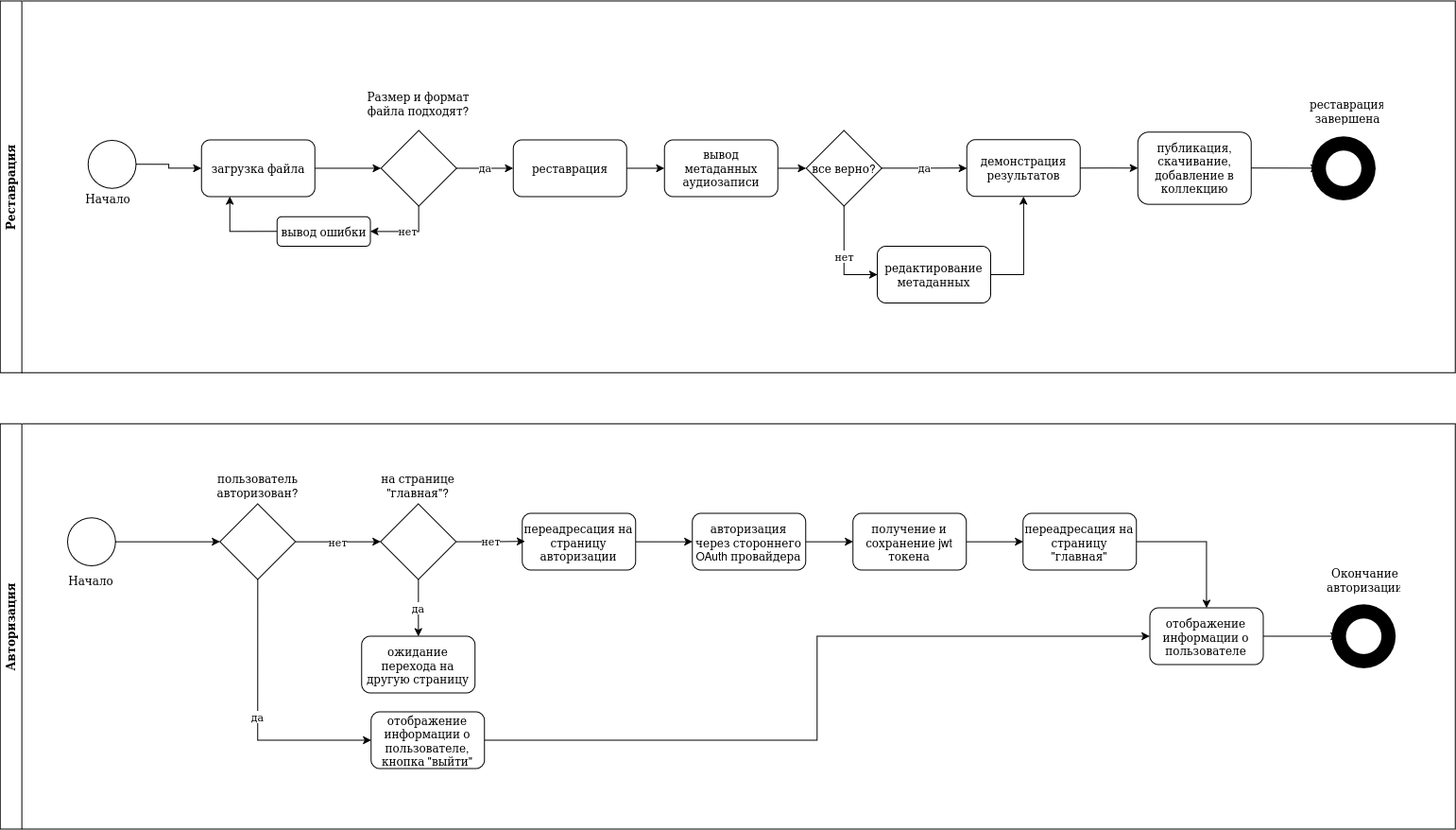
В итоге, технологическая часть описывает выбор и сравнение инструментов для реализации фронтенд-составляющей сервиса «Голос победы». Проведенный анализ подтверждает целесообразность использования Vue.js, Pinia, Tailwind CSS, jsmediatags и Axios, которые обеспечивают создание интуитивного, адаптивного и эффективного интерфейса.

## 2.6 Разработка проекта решений

В данном разделе будет подробно описан планируемый проект сервиса рестоврации аудиозаписей, включая схемы и диаграммы

**Рисунок 2.2.1 — UseCase диаграмма**

Модель потоков действий по ролям представлена в UseCase диаграмме(рис. 2.2.1)

**Рисунок 2.2.2 - Диаграмма процессов**

Для визуального моделирования этапов бизнес-процесса построена диаграмма BPMN. Основной бизнес-процесс «Реставрация» изображён на диаграмме (рис. 2.2.2).

# 3 ПРАКТИЧЕСКАЯ ЧАСТЬ

В рамках главы будет подробно рассмотрен процесс разработки фронтенд составляющей информационной системы. Будет описана реализация SPA (Single Page Application) с использованием компонентной архитектуры на базе фреймворка Vue.js.

В процессе реализации была разработана файловая архитектура, изображенная далее:

*Листинг 3.1 — Файловая архитектура проекта*

src

├── App.vue

├── assets

│   ├── avatar-placeholder.png

│   ├── base.css

│   ├── input.css

│   ├── logo.svg

│   └── main.css

├── components

│   ├── AudioVisualizer

│   │   ├── AudioVisualizer.vue

│   │   └── MiniVisualizer.vue

│   ├── Header

│   │   ├── AnimationToggle.vue

│   │   ├── DesktopNav.vue

│   │   ├── Header.vue

│   │   ├── ThemeSwitcher.vue

│   │   └── TopUserProfile.vue

│   ├── Layout

│   │   ├── BlankLayout.vue

│   │   └── MainLayout.vue

│   ├── Music

│   │   ├── MobileMusicBar.vue

│   │   ├── MusicBar.vue

│   │   └── QueuePopOver.vue

│   ├── MusicSidebar

│   │   ├── MusicSidebar.vue

│   │   └── QueueSidebar.vue

│   ├── Sidebar

│   │   ├── NavMenu.vue

│   │   ├── SearchInput.vue

│   │   └── Sidebar.vue

│   ├── StepIndicator.vue

│   ├── steps

│   │   ├── Step1Content.vue

│   │   ├── Step2Content.vue

│   │   ├── Step3Content.vue

│   │   ├── Step4Content.vue

│   │   └── Step5Content.vue

│   └── TrackLayout

│   ├── TrackCardLibrary.vue

│   ├── TrackCard.vue

│   └── TrackRow.vue

├── main.js

├── router

│   └── index.js

├── services

│   ├── api.js

│   ├── authservice.js

│   ├── avatarService.js

│   └── trackService.js

├── store

│   ├── auth.js

│   ├── music.js

│   ├── parser.js

│   ├── player.js

│   ├── restoration.js

│   ├── theme.js

│   ├── track.js

│   └── visualizer.js

└── views

├── Callback.vue

├── Collection.vue

├── Home.vue

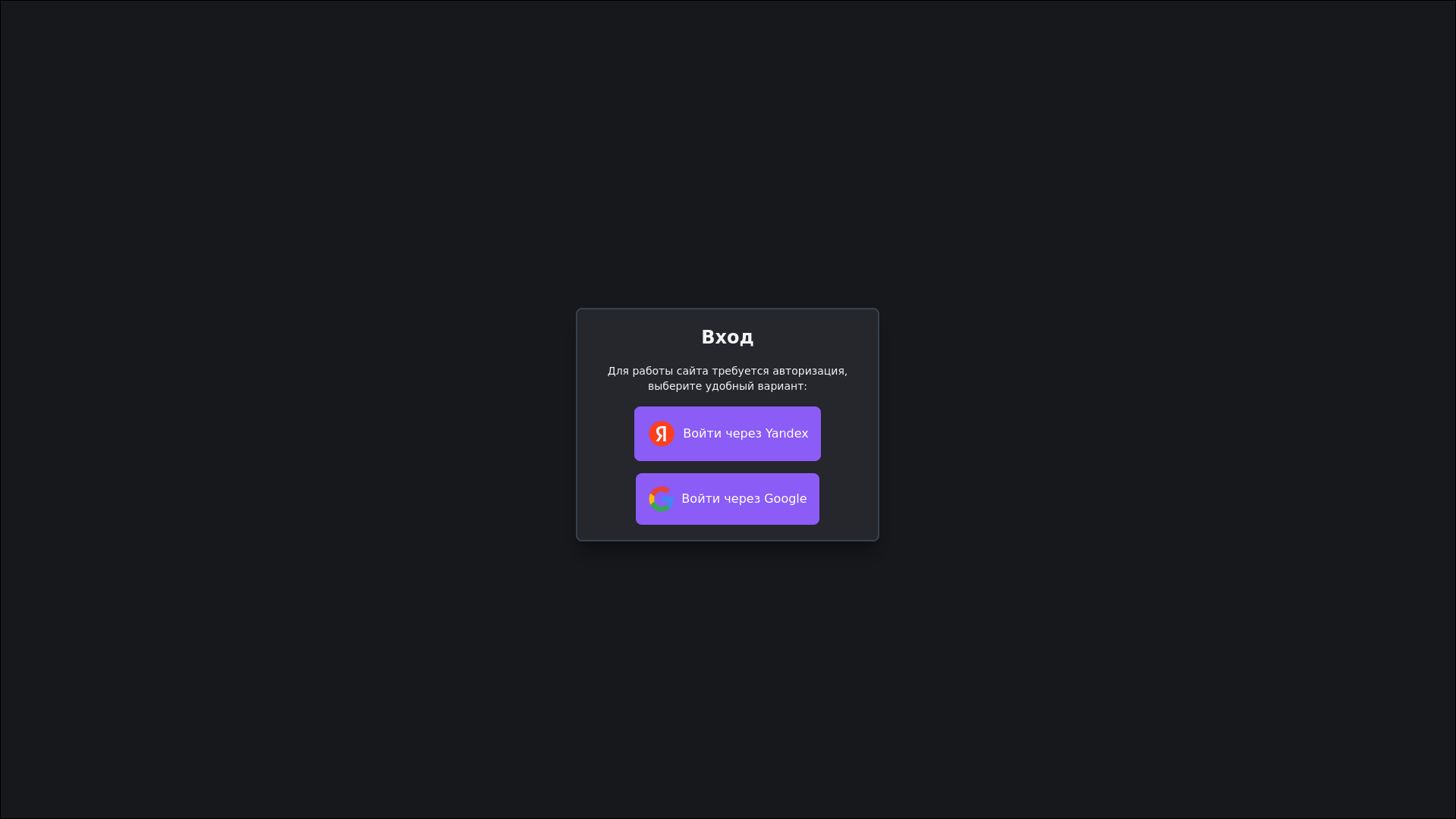
├── Library.vue

├── Login.vue

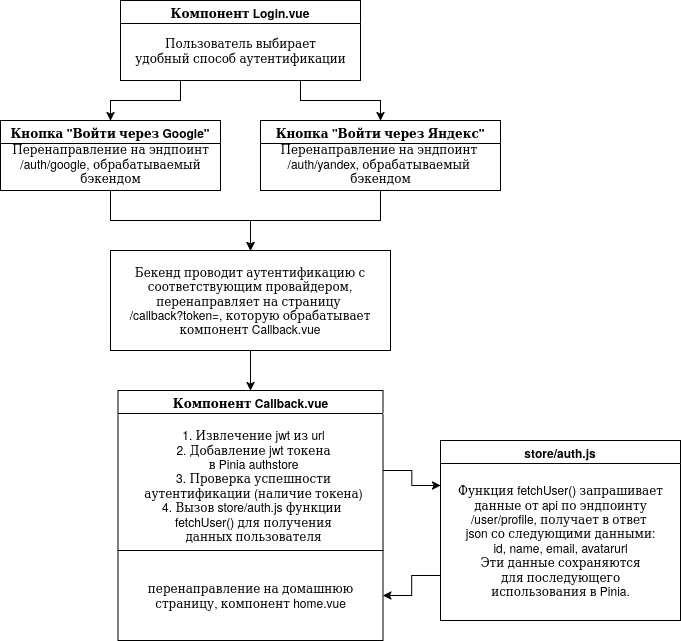
└── Restoration.vue

## 3.1 Разработка аутентификации

В процессе реализации было решено, что аутентификация будет проводиться через сторонних OAuth провайдеров, а именно, для удобства конечного пользователя были выбраны сервисы аутентификации Yandex и Google. Таким образом, требуется создать страницу с выбором предпочитаемого провайдера, в нашем случае это будет компонент Login.vue (см. Листинг 3.1).

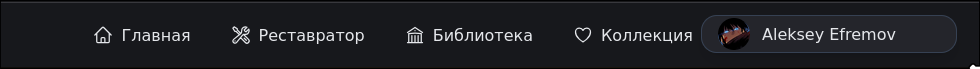
**Рисунок 3.1.1 — Страница авторизации, компонент Login.vue**

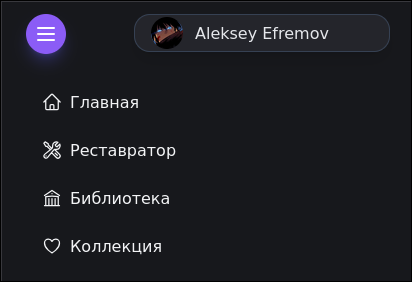
Авторизация через сторонних провайдеров и сохранение сессии пользователя основано на технологии jwt-токена который сохраняется в cookie файлы браузера благодаря технологии Pinia хранилища для аутентификационных данных.

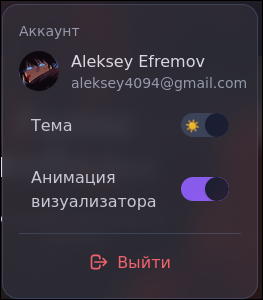
**Рисунок 3.1.2 — Логика системы аутентификации**

## 3.2 Разработка общих компонентов

Общие компоненты не зависят от страницы, на которой в данный момент находится пользователь. Они должны отвечать требованиям адаптивности, потому многие из них меняются относительно ширины используемого экрана. Это достигается при помощи специальных идентификаторов tailwindcss, а также использованием идентификатора «@media» в пользовательских css стилях.

**Рис. 3.2.1 Компонент header для широких экранов**

**Рис.3.2.2 Компонент header в «развернутом» положении для мобильных устройств**

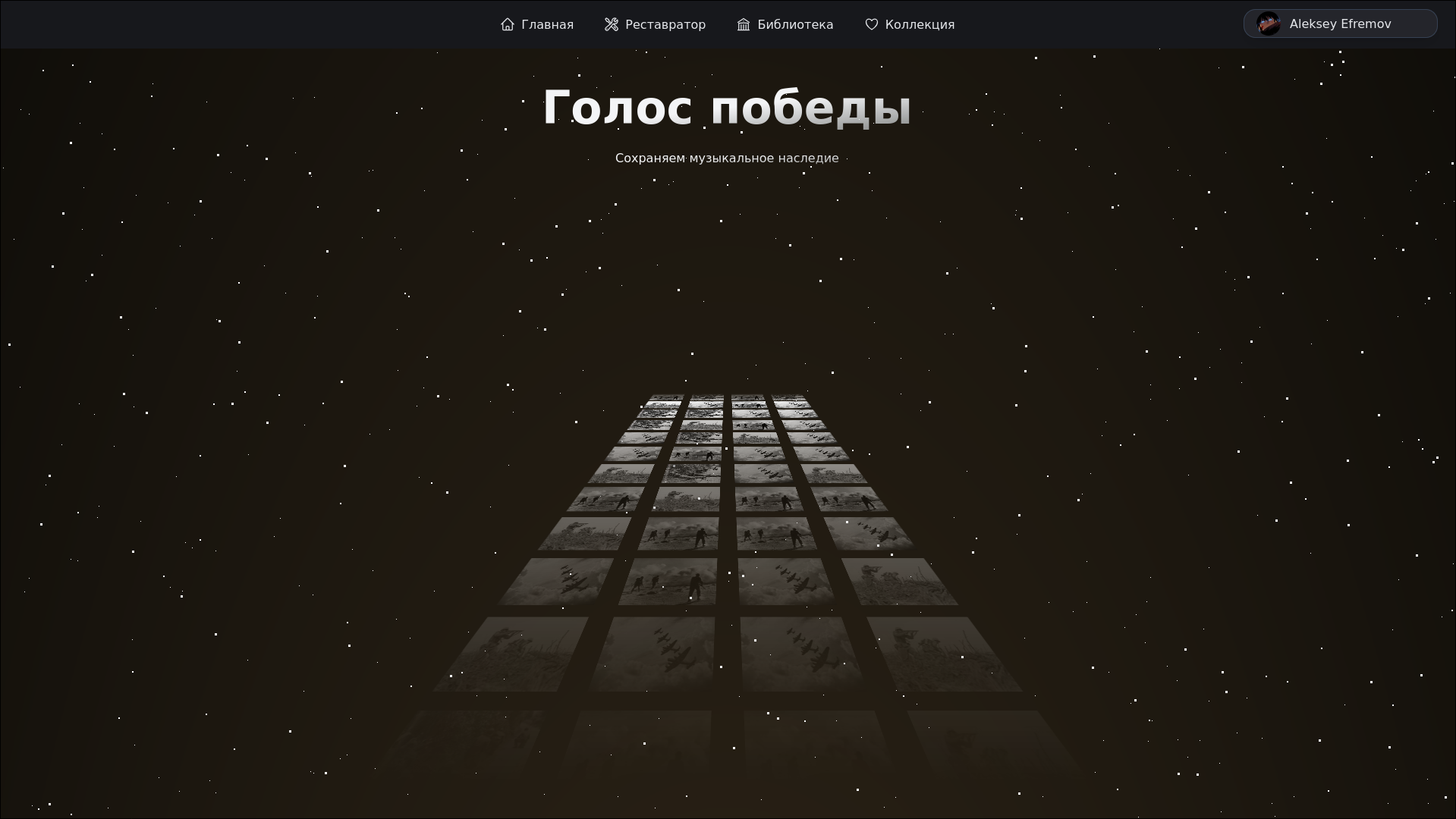
**Рис.3.2.3 Компонент UserProfile в «развернутом положении**

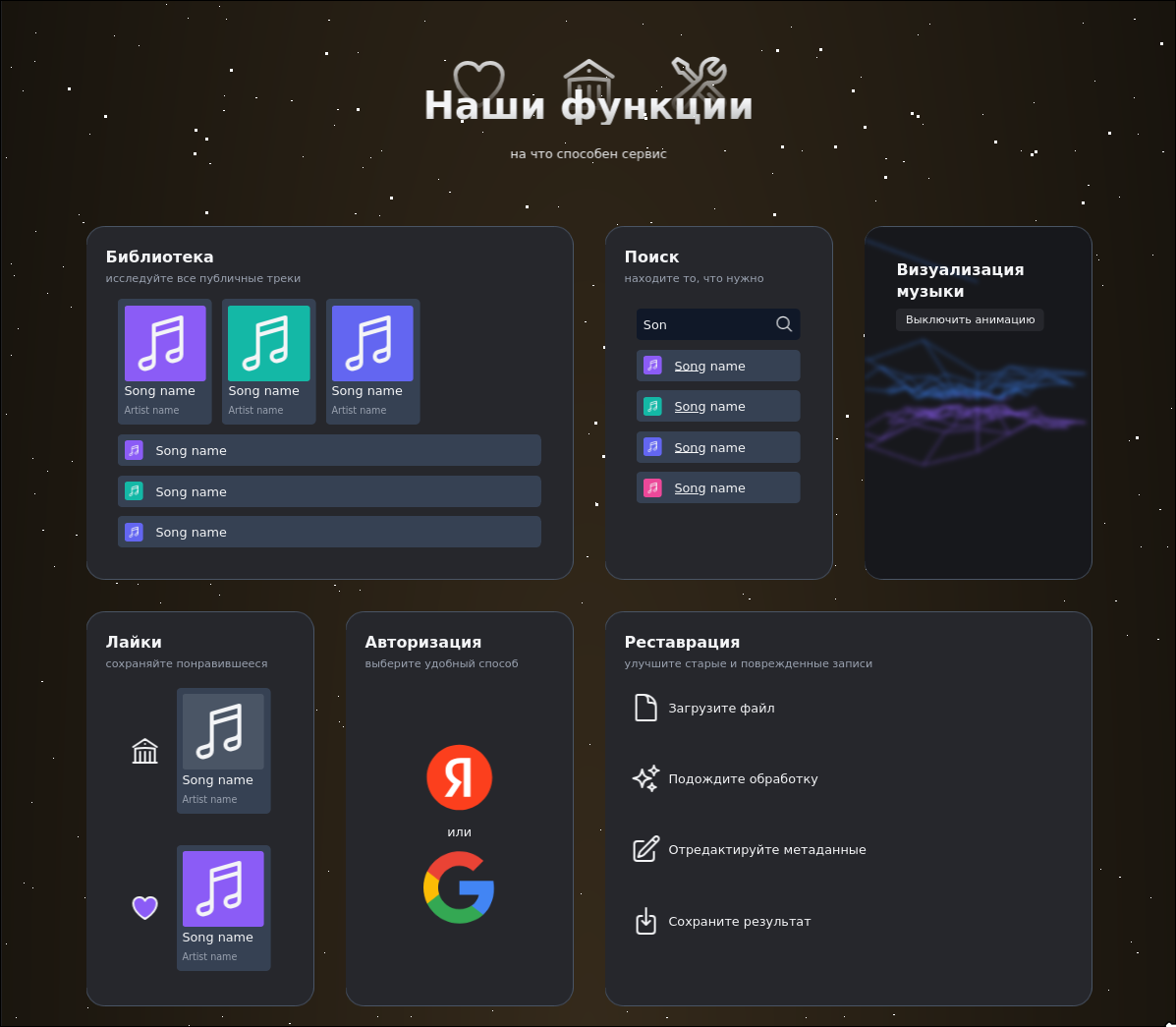
Компонент UserProfile показывает актуальную информацию о пользователе из Pinia AuthStore, а также имеет переключатели препочитаемой цветовой темы сайта и включения анимации визуализатора музыки. Функционал двух цветовых тем реализован с помощью использования идентификаторов «dark:» и «light:» технологии tailwindcss.

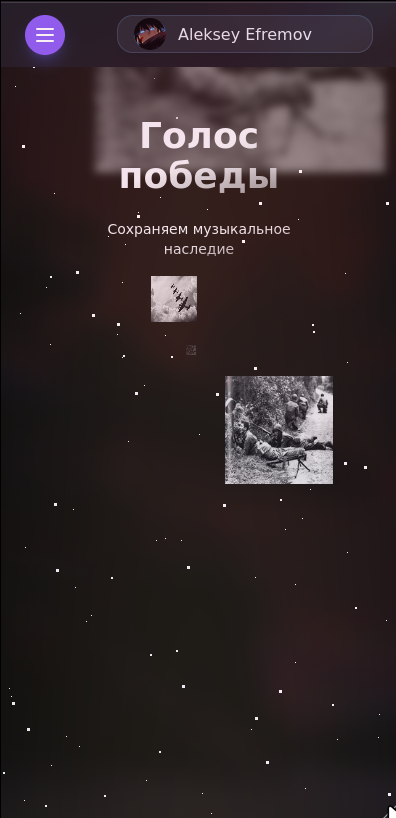
## 3.3 Разработка ключевых страниц сайта.

Исходя из требований к сервису было создано 4 страницы сайта:

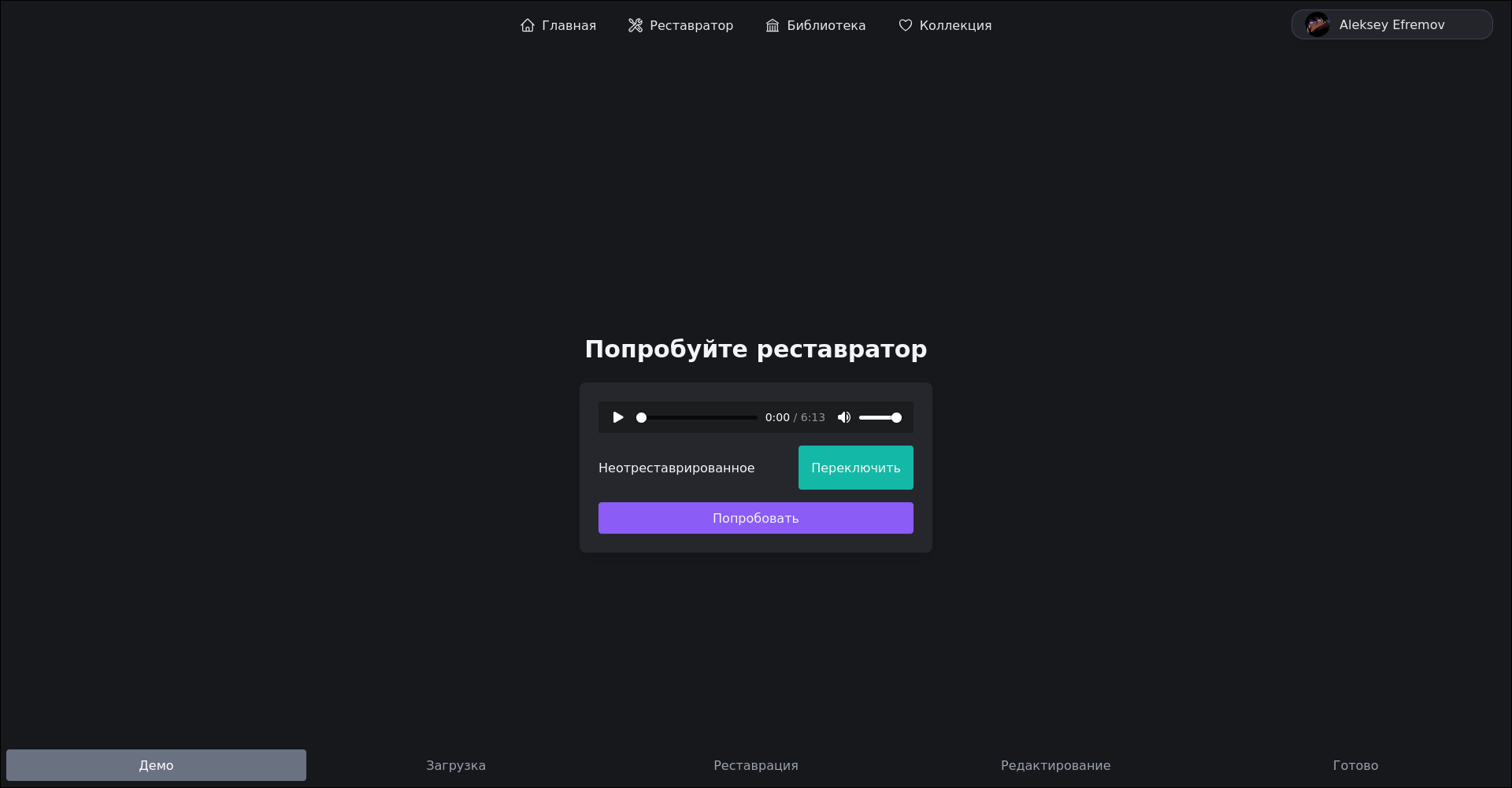
* Главная — «лицо» сервиса, список функционала
* Реставратор — функционал реставрирования музыки
* Библиотека — публичные аудиофайлы, выгруженные пользователями.
* Коллекция — личная коллекция пользователя, а также добавленные в нее из библиотеки композиции.

**Рис. 3.3.1 — страница «Главная», компонент Home.vue, часть 1**

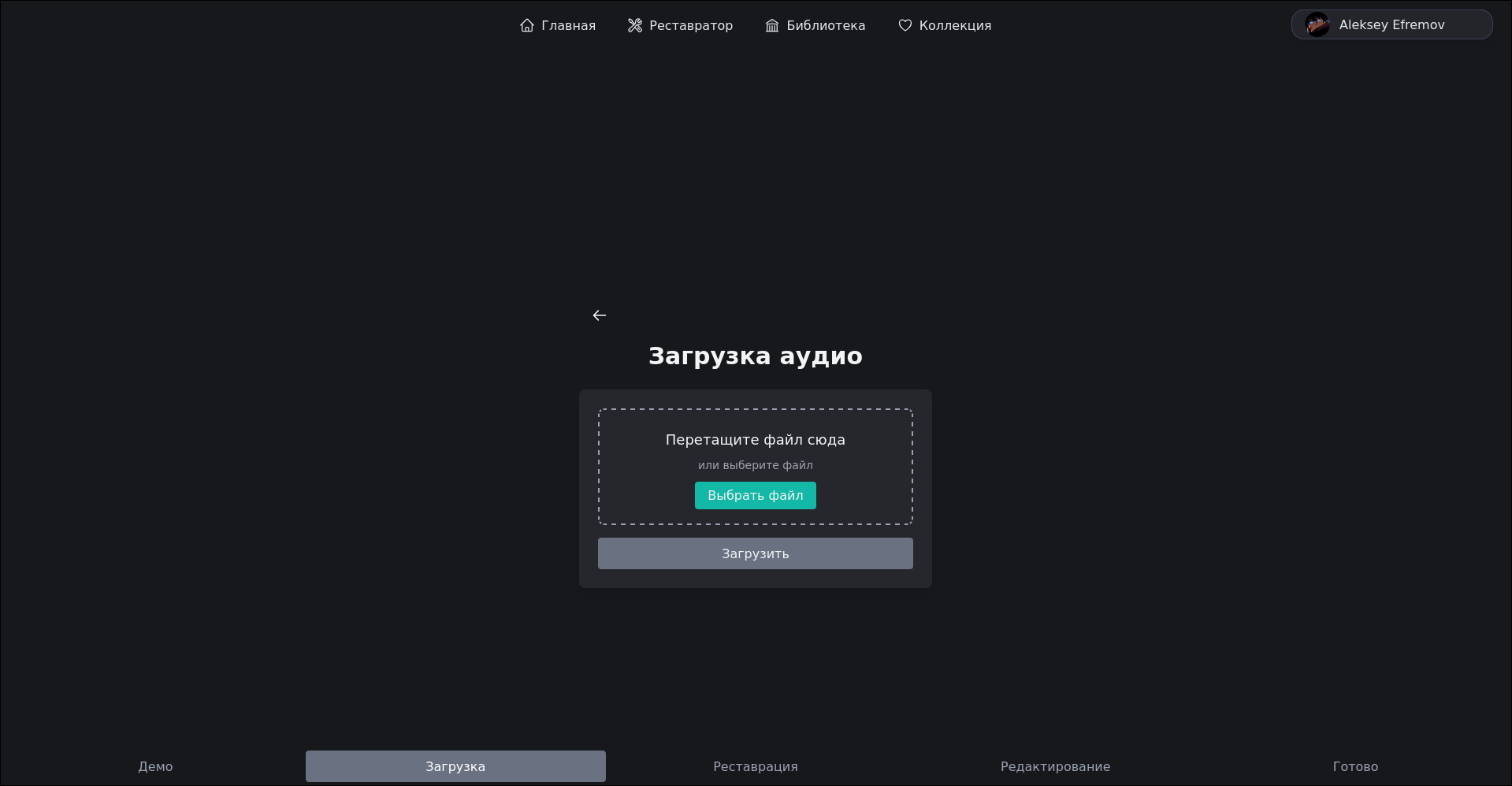
**Рис. 3.3.2 — страница «Главная», компонент Home.vue, часть 2**

**Рис. 3.3.3 — Анимация на странице «Главная» для мобильных устройств**

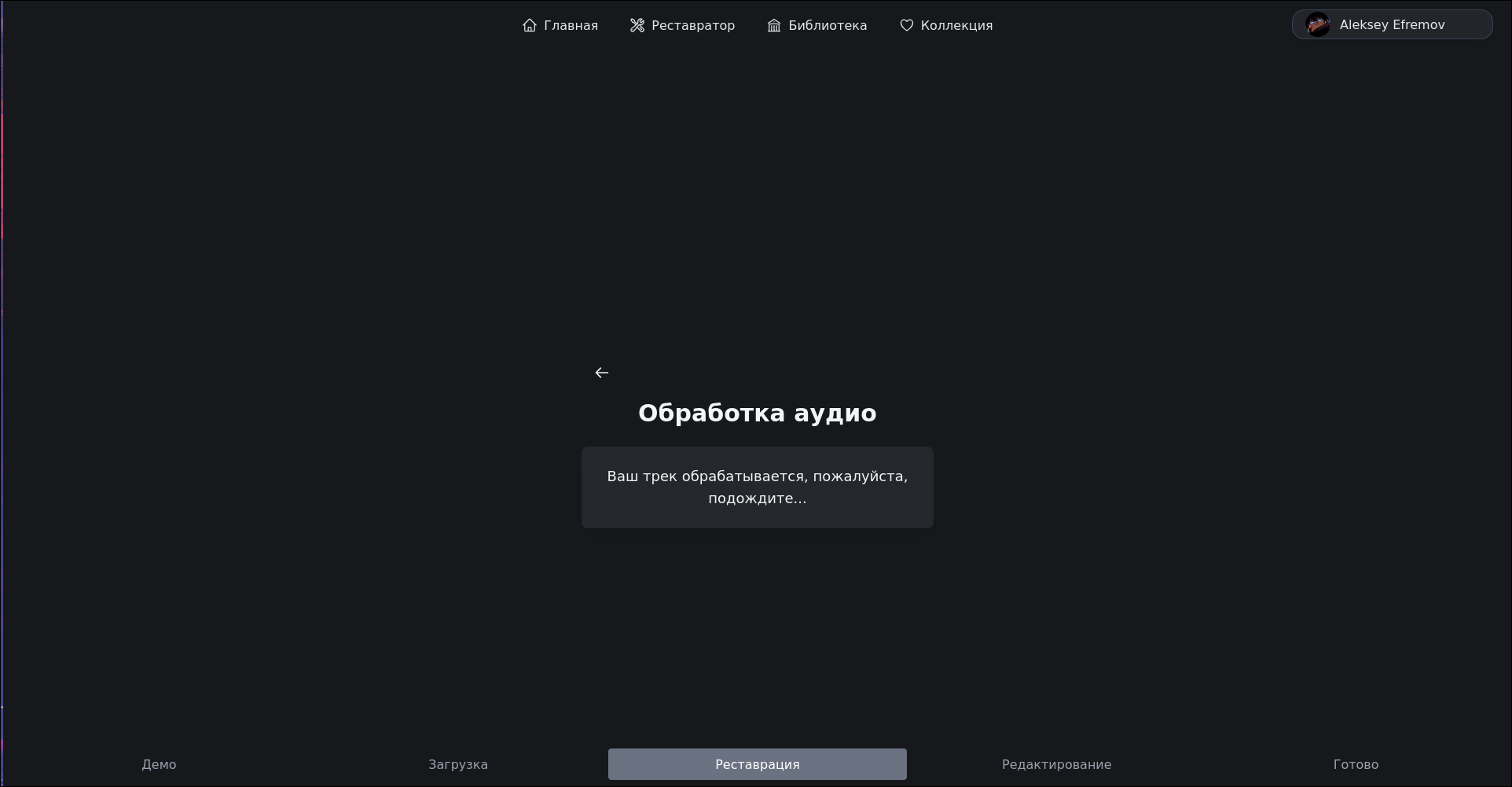
Страница «реставрация» имеет пошаговую логику, для этого существуют компоненты Step1Content.vue … Step5Content.vue, для каждого из шагов, а также компонент Restoration.vue, который отвечает за логику переключения этих шагов.

**Рис. 3.3.4 — Страница «Реставрация», шаг 1**

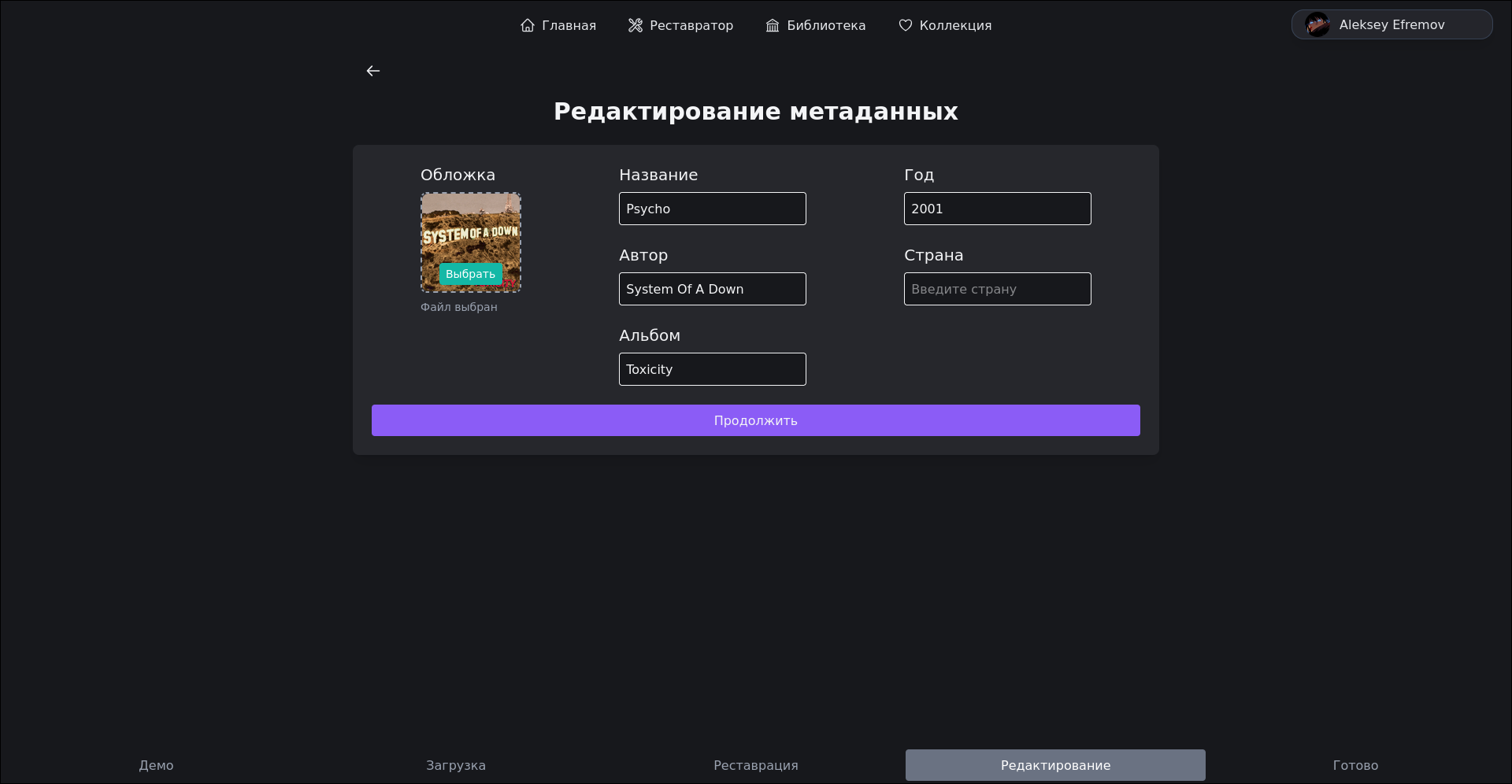
На шаге 1 страницы «Реставрация» представлена демонстрация возможностей реставрации, а также возможность перехода на следующий шаг.

**Рис. 3.3.5 — Страница «Реставрация», шаг 2**

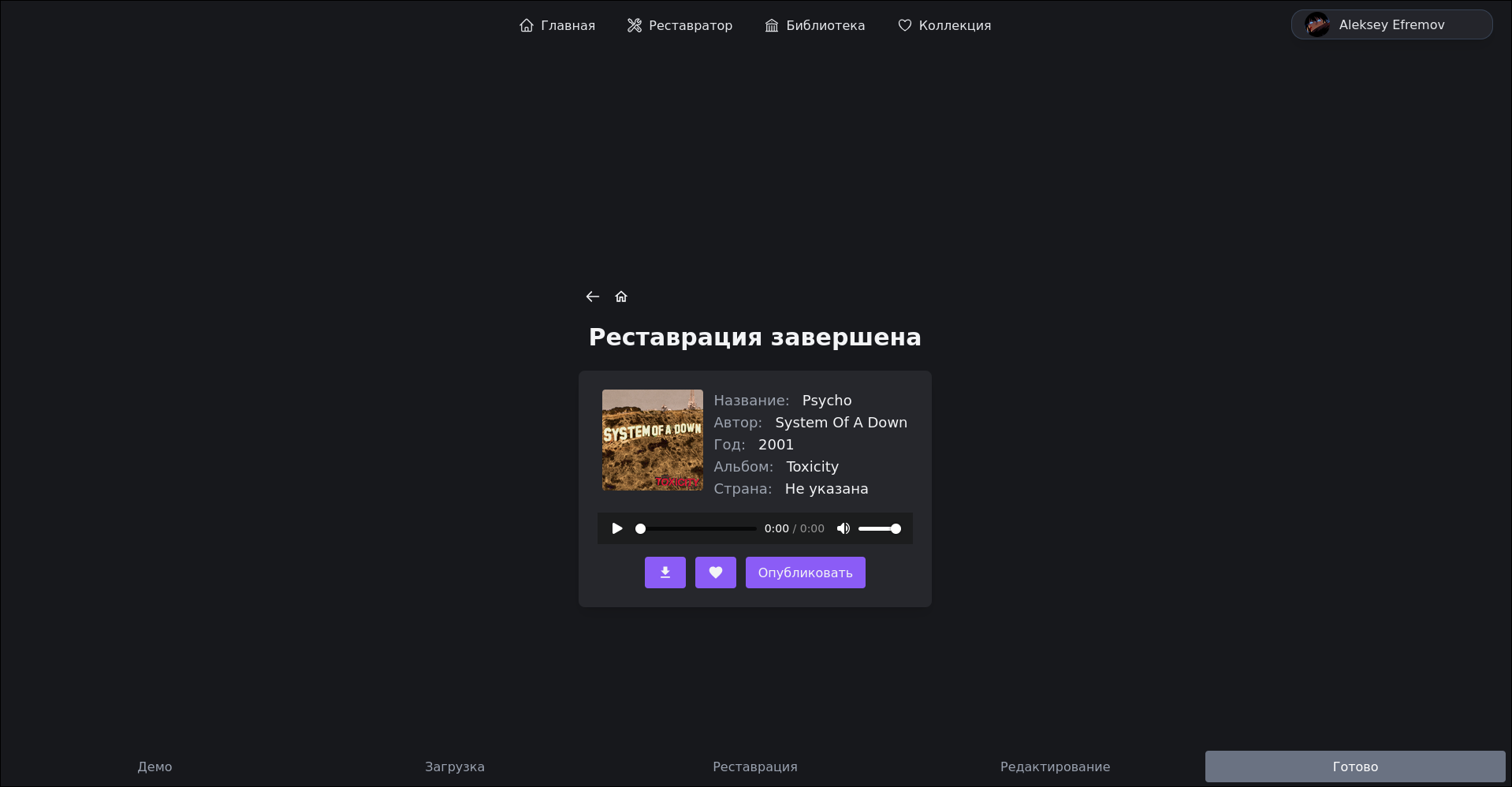
На шаге 2 страницы «Реставрация» реализован функционал загрузки файлов методом выбора файла или «drag&drop». После загрузки файла происходит обработка метаданных библиотекой jsmediatags. Метаданные сохраняются в Pinia store/restoration.js, а файл отправляется по https на бэкенд эндпоинт «/restoration/upload» в формате «multipart/formData».

**Рис. 3.3.6 — Страница «Реставрация», шаг 3**

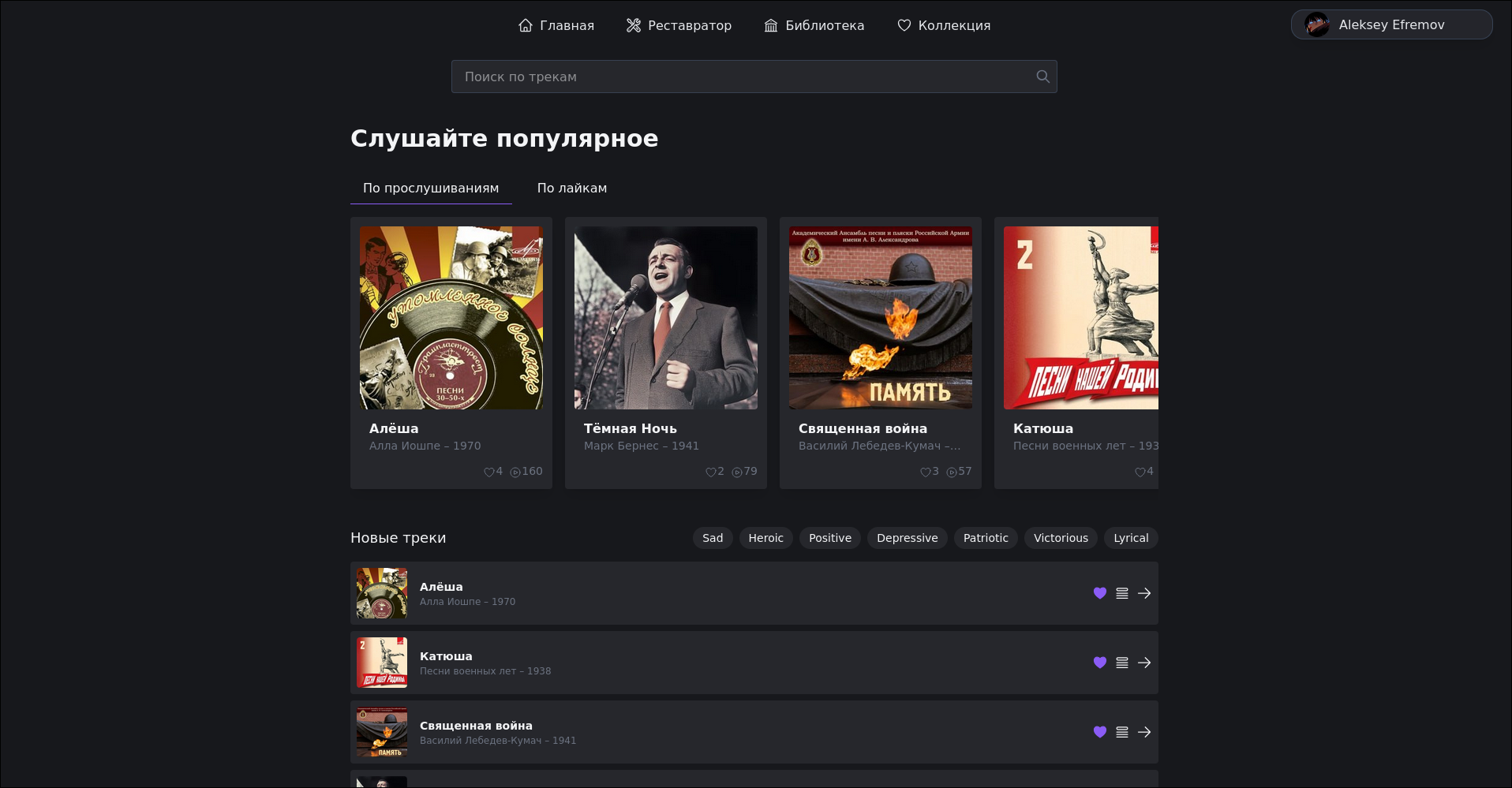
На шаге 3 пользователю показывается сообщение об обработке, реставрации пользовательского аудиофайла. Переход на следующий шаг реализован при помощи периодического отправления запросов на эндпоинт «/restoration/isReady», который сигнализирует о том, когда обработка ML-пайплайна закончится.

**Рис. 3.3.7 — страница «Реставрация», шаг 4**

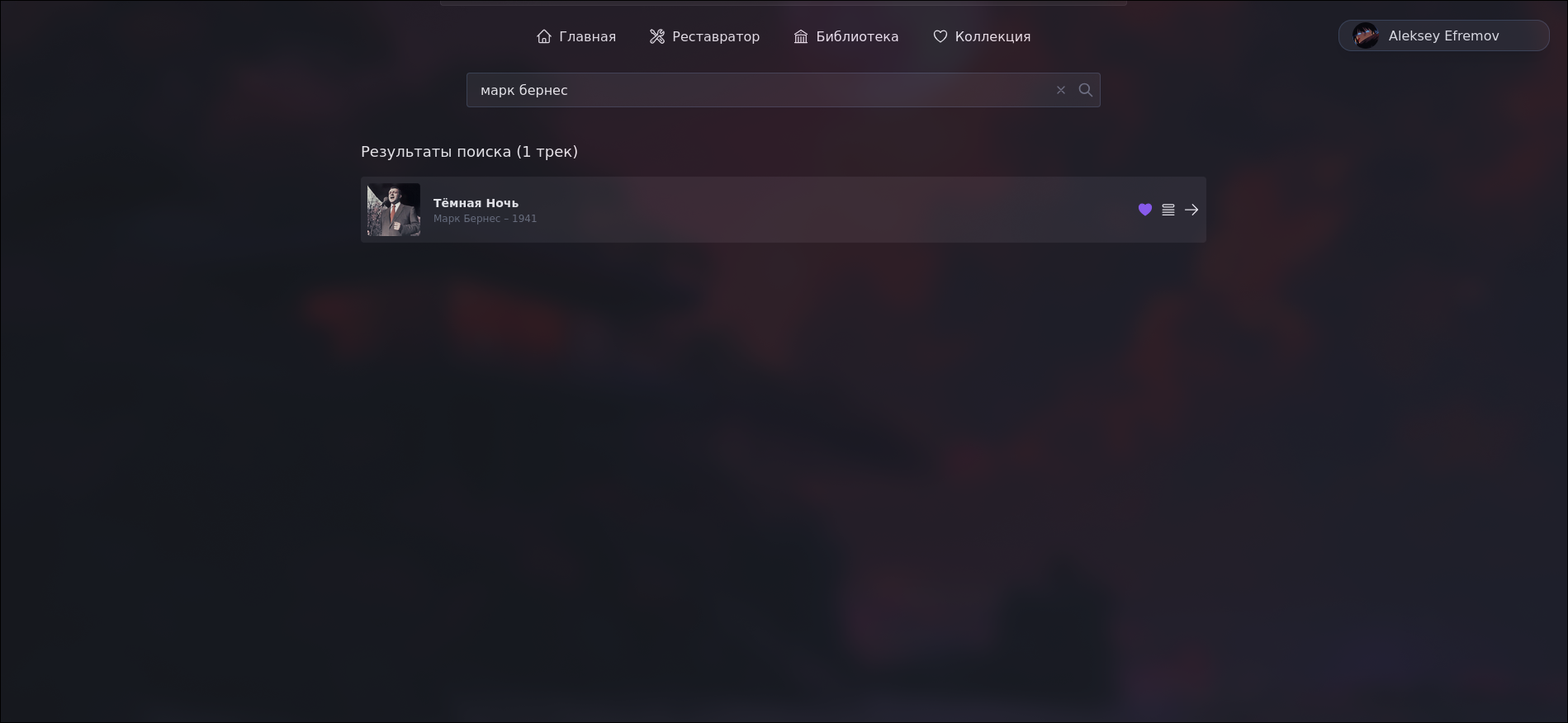
На шаге 4 представлен функционал редактирования пользователем метаданных, которые были извлечены на шаге 2. Для обложек треков используется конвертация в формат base64.

**Рис. 3.3.8 — страница «Реставрация», шаг 5**

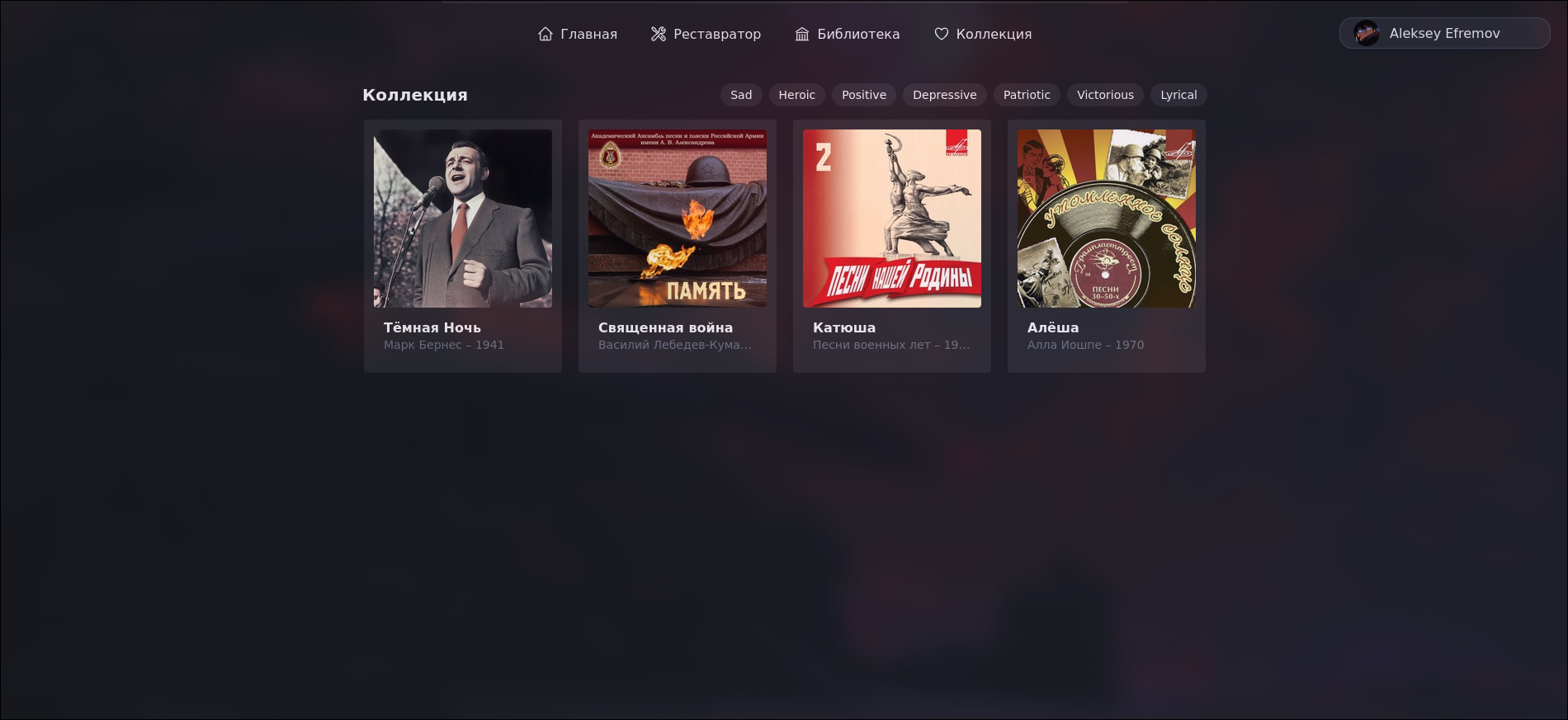
Последний, пятый шаг страницы «реставрации» представляет собой предпросмотр данных аудиозаписи, функционал прослушивания отреставрированной версии и последующее скачивание, добавление в коллекцию и публикация в библиотеку.

**Рис.3.3.9 — Страница «Библиотека»**

На странице представлено 3 секции: поиск, популярное, и новые треки. Раздел «популярное» получает список композиций с эндпоинта «/public-library/top-plays» и «/public-library/top-likes». Раздел «Новые треки» получает список аудиофайлов по эндпоинту «/public-library», добавляя в body запроса список тегов, которые выбрал пользователь.

**Рис. 3.3.10 — Страница «Библиотека», поиск треков.**

Поиск композиций происходит с помощью отправки запроса на эндпоинт «/search», который возвращает список найденных аудиофайлов при помощи технологии ElasticSearch.

**Рис. 3.3.11 — Страница «Коллекция»**

На последней странице, «Коллекция», представлены добавленные пользователем композиции, получаемые по эндпоинту «user/library/list», с выбранными тегами в теле запроса.

## 3.4 Разработка функционала прослушивания.

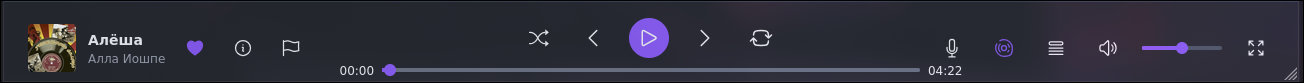
Как было представлено ранее, в сервисе есть 2 компонента, отвечающие за отображение информации о композиции — это TrackCard.vue и TrackRow.vue.

**Рис. 3.4.1 — Представление списка из нескольких TrackCard**

**Рис. 3.4.2 — Представление списка из несколькихх TrackRow.**

Оба компонента имеют функционал проигрывания композиции, добавление в коллекцию, или же «отметки нравится», добавление в конец очереди и добавление в начало очереди, «играть следующим». В случае TrackCard данный функционал отображается при наведении кусора мыши на компонент, эффект hover.

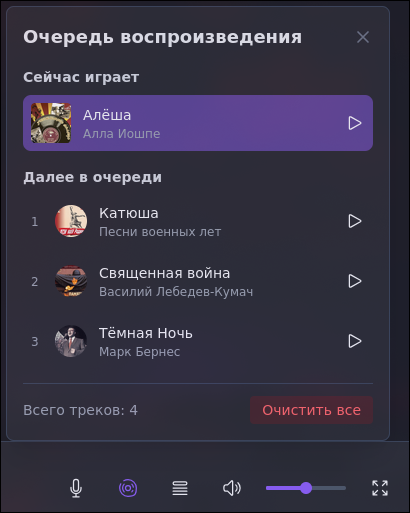
Музыкальный плеер находится в нижней части экрана. Было реализовано несколько его состояний в зависимости от ширины экрана пользователя.

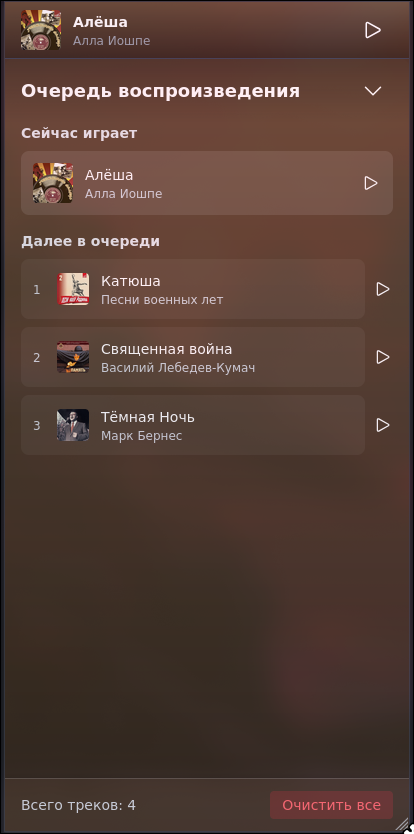
**Рис. 3.4.3 — Аудиоплеер для широких экранов**

**Рис. 3.4.4 — Аудиоплеер для мобильных дисплеев**

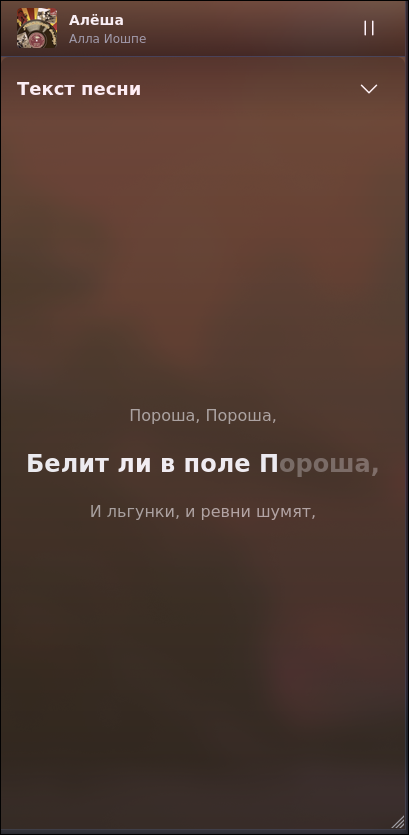
**Рис. 3.4.5 — Развернутая версия мобильного плеера**

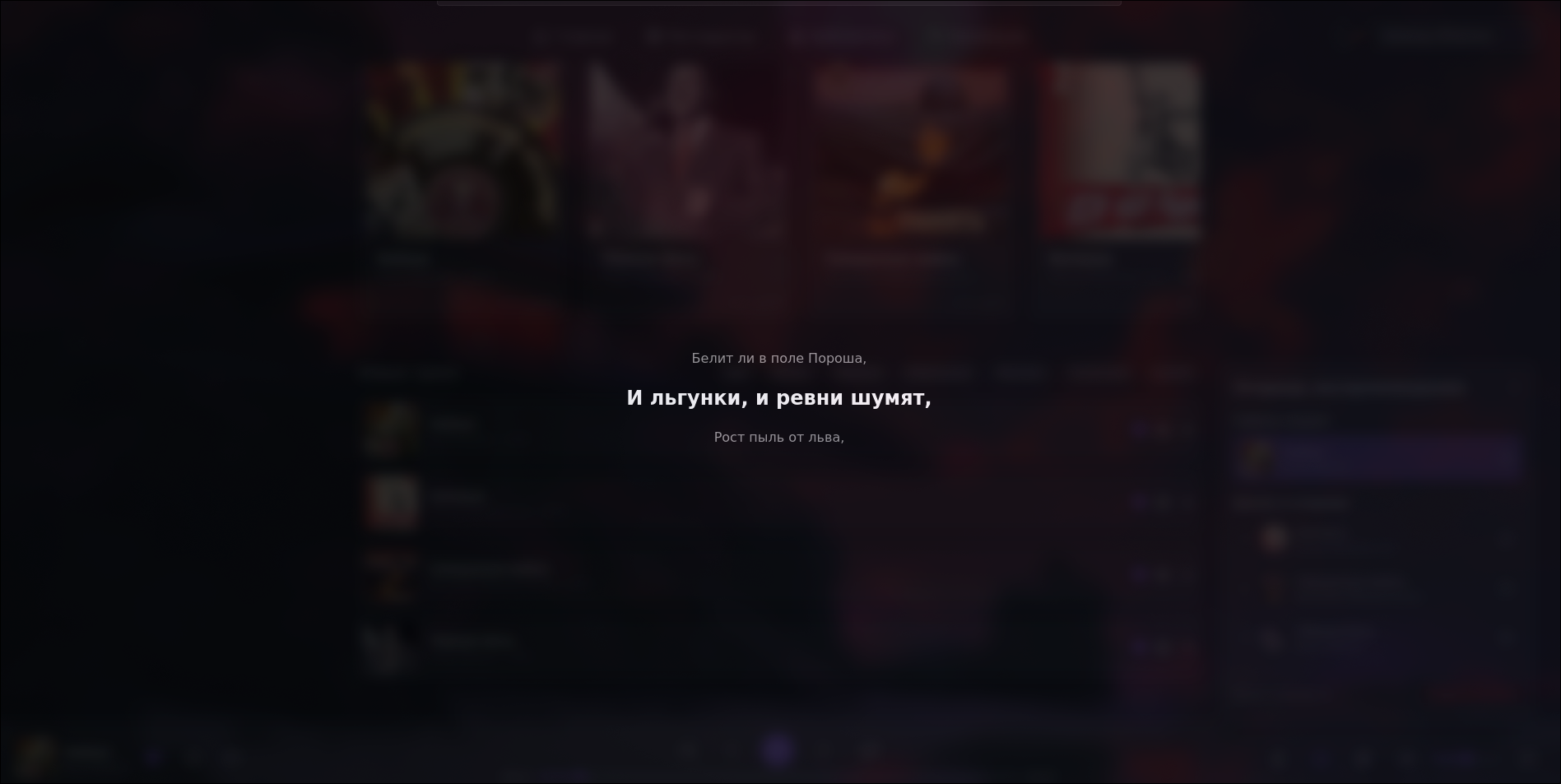
Плеер имеет функционал прослушивания, перемотки, паузы. Загрузка аудиофайла проходит по эндпоинту «/restoration/stream/${trackId}?version=${version}», с учетом версии трека — отреставрированной или же нет. За переключение версии отвечает иконка в виде музыкальной пластинки.

**Рис. 3.4.6 — Модальное окно очереди прослушивания для широких дисплеев**

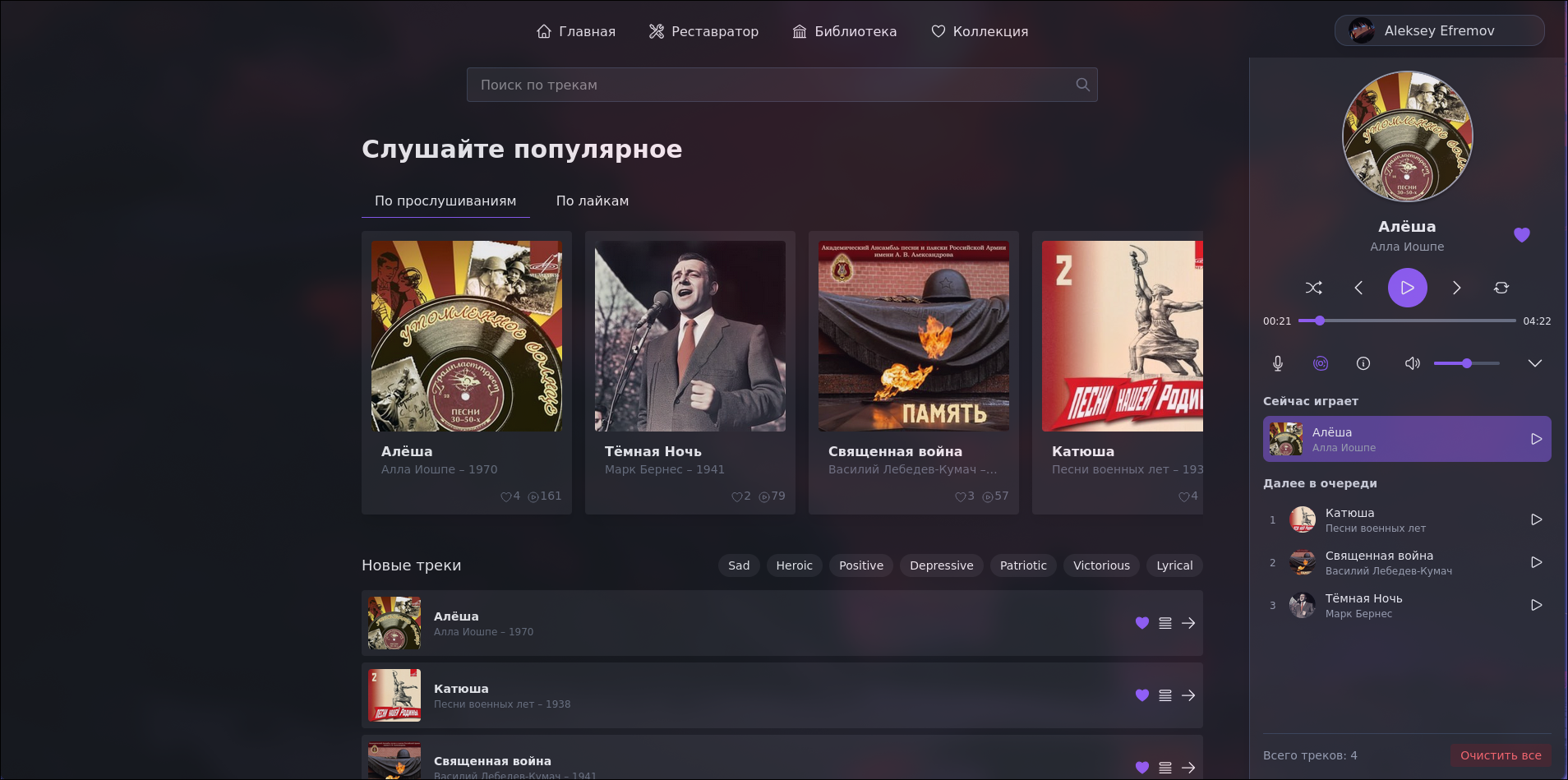
**Рис. 3.4.7 — Окно очереди воспроизведения для мобильных устройств**

В компонентах очереди воспроизведения композиций реализован функционал «drag&drop»

**Рис. 3.4.8 — Просмотр текста произведения для мобильных устройств.**

**Рис. 3.4.9 — Просмотр текста произведения для широких дисплеев.**

Также для широких дисплеев был реализовано состояние музыкального плеера в формате «сайдбара», то есть окна справа области сайта.

**Рис. 3.4.10 — Музыкальной плеер боковой панели для широких дисплеев.**

За состояние плеера «боковая панель» отвечает группа компонентов «MusicSidebar».

## 3.5 Инструкция по эксплуатации

****3.5.1 Клиентский сценарий взаимодействия****

**Сервис «Голос победы» предоставляет удобный интерфейс для реставрации аудиозаписей и управления музыкальным контентом. Основные действия пользователя включают:**

**- **Аутентификация.** Для доступа к личной коллекции пользователь может зарегистрироваться или войти через Google или Yandex, выбрав провайдера на странице авторизации.**

**- **Загрузка аудиофайлов**. На странице реставрации пользователь загружает MP3-файл через выбор или перетаскивание. После загрузки отображаются метаданные (название, автор), которые можно подтвердить или изменить.**

**- **Реставрация**. После отправки файла отображается сообщение об ожидании. По завершении обработки пользователь может отредактировать метаданные, прослушать результат и скачать, опубликовать или добавить трек в коллекцию.**

**- **Библиотека** и коллекция. Публичная библиотека позволяет просматривать и искать аудиозаписи по тегам. Личная коллекция дает возможность управлять добавленными треками.**

**- **Прослушивание**. Встроенный плеер поддерживает воспроизведение, паузу, перемотку и переключение версий трека. Очередь воспроизведения настраивается через интерфейс.**

**Интерфейс адаптирован для любых устройств вне зависимости от ширины экрана.**

****3.5.2 Административный сценарий взаимодействия****

**Администрирование осуществляется через платформу Hasura с GraphQL API. Администратор использует консоль Hasura для:**

**- Управления записями и метаданными (просмотр, редактирование, удаление).**

**- Мониторинга активности пользователей.**

**- Обработка жалоб на композиции.**

**Для работы администратор авторизуется и выполняет GraphQL-запросы. Hasura обеспечивает гибкость и производительность.**

**В итоге, клиентский сценарий упрощает реставрацию и прослушивание аудио, а Hasura поддерживает эффективное администрирование.**

# ЗАКЛЮЧЕНИЕ

Курсовая работа посвящена разработке фронтенд-составляющей сервиса «Голос победы», предназначенного для автоматизации реставрации аудиозаписей, связанных с Великой Отечественной войной. В ходе анализа предметной области выявлена значимость сохранения исторических аудиоматериалов и ограничения ручной реставрации, такие как сложность и высокая стоимость. Сравнение существующих решений (iZotope RX, Adobe Audition, Audacity, Audio Restoration Online) показало, что ни одно из них не сочетает веб-доступность, простоту и функции музыкального сервиса, что обосновало актуальность проекта.

В итоге разработана фронтенд-составляющая, обеспечивающая удобный интерфейс для загрузки, обработки и управления аудиозаписями. Сервис упрощает доступ к реставрации для широкой аудитории, включая музеи и частных пользователей, и способствует сохранению культурного наследия. Перспективы развития включают добавление новых форматов аудио и оффлайн-режима. Все поставленные задачи выполнены, а результаты подтверждают социальную значимость проекта.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Vue.js / [Электронный ресурс] // Vue.js : [сайт]. — URL: <https://vuejs.org/> (дата обращения: 09.06.2025).
2. Pinia / [Электронный ресурс] // Pinia : [сайт]. — URL: <https://pinia.vuejs.org/> (дата обращения: 09.06.2025).
3. Tailwind CSS / [Электронный ресурс] // Tailwind CSS : [сайт]. — URL: <https://tailwindcss.com/> (дата обращения: 09.06.2025).
4. jsmediatags / [Электронный ресурс] // GitHub : [сайт]. — URL: <https://github.com/aadsm/jsmediatags> (дата обращения: 09.06.2025).
5. Hasura / [Электронный ресурс] // Hasura : [сайт]. — URL: <https://hasura.io/docs/latest/> (дата обращения: 09.06.2025).
6. 10 common uses for audio restoration in mastering / [Электронный ресурс] // iZotope : [сайт]. — URL: <https://www.izotope.com/en/learn/10-common-uses-for-audio-restoration-in-mastering.html> (дата обращения: 09.06.2025).
7. What Is Audio Restoration? Tips, Tricks, and More / [Электронный ресурс] // Boris FX : [сайт]. — URL: <https://borisfx.com/blog/what-is-audio-restoration-tips-tricks-more/> (дата обращения: 09.06.2025).
8. Сравнение JS-фреймворков: React, Vue и Hyperapp / [Электронный ресурс] // Хабр : [сайт]. — URL: <https://habr.com/ru/companies/ruvds/articles/417483/> (дата обращения: 07.05.2025).
9. React vs Vue – подробное сравнение и перспективы / [Электронный ресурс] // Хабр : [сайт]. — URL: <https://habr.com/ru/articles/904698/> (дата обращения: 09.05.2025).
10. Как выбрать фронтенд-фреймворк: сравнение React, Vue и Angular / [Электронный ресурс] // RuWeb : [сайт]. — URL: <https://ruweb.net/articles/kak-vybrat-frontend-frejmvork-sravnenie-react-vue-i-angular> (дата обращения: 14.05.2025).
11. BPMN / [Электронный ресурс] // Ensi : [сайт]. — URL: <https://docs.ensi.tech/analyst-guides/tools/diagrams/bpmn> (дата обращения: 16.05.2025).