Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Модели данных и системы управления базами данных

*К защите допустить*:

И. О. Заведующего кафедрой

информатики

\_\_\_\_\_\_\_\_\_\_\_\_ С. И. Сиротко

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

**ПРОГРАММНОЕ СРЕДСТВО, РЕАЛИЗУЮЩЕЕ МУЗЫКАЛЬНЫЙ СЕРВИС ДЛЯ АВТОРОВ ПЕСЕН И СЛУШАТЕЛЕЙ**

БГУИР КП 1-40 04 01 003 ПЗ

Студент: К. А. Антилевская

Руководитель: А. В. Давыдчик

Минск 2024**СОДЕРЖАНИЕ**

Введение 3

1 Описание предметной области, постановка задачи

2 Обоснование выбора СУБД PostgreSQL, описание возможностей СУБД

1 PostgreSQL

3 Проектирование базы данных в выбранной СУБД

4 Разработка клиентского приложения

[Заключение](#_Toc151561179)

[Список использованных источников](#_Toc151561180)

**ВВЕДЕНИЕ**

В современном цифровом мире, где доступ к информации и скорость обработки данных играют все более важную роль, эффективное использование баз данных становится неотъемлемым элементом многих приложений. Базы данных служат инструментом для организации информации и обеспечения доступа к ней во всех сферах деятельности, начиная от предприятий и исследовательских организаций и заканчивая личным использованием.

База данных – это упорядоченный набор структурированной информации или данных, обычно хранящийся и управляемый с использованием специализированного программного обеспечения, называемого системой управления базами данных (СУБД).

База данных может содержать данные различных типов, такие как текст, числа, изображения, аудиофайлы и многое другое. Данные обычно хранятся в виде строк и столбцов, формирующих таблицу. Этими данными можно легко управлять, изменять, обновлять, контролировать и упорядочивать. В большинстве баз данных для записи и запросов данных используется язык структурированных запросов (SQL). SQL – это язык программирования, используемый в большинстве реляционных баз данных для запросов, обработки и определения данных, а также контроля доступа [1].

Одним из примеров приложений, требующих эффективного хранения и управления информацией, является музыкальный сервис. Музыкальная индустрия, подобно многим другим, вынуждена приспосабливаться к современным запросам пользователей и рынка. Сервисы стриминга музыки позволяют пользователям искать, прослушивать и создавать плейлисты с миллионами композиций из различных жанров.

Создание и поддержание музыкальных сервисов также требует не только удобного пользовательского интерфейса, но и мощной инфраструктуры для обработки и хранения данных о музыкальных треках, альбомах, плейлистах. Для эффективного управления этой информацией используются базы данных, которые обеспечивают надежное хранение, организацию и быстрый доступ к данным.

Основная задача курсового проекта – разработать реляционную базу данных «Музыкальный сервис для авторов песен и слушателей» и пользовательское приложение для работы с составленной базой данных. В ходе разработки решается еще одна задача курсового проекта – закрепление практических навыков разработки и проектирования баз данных.

**1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ, ПОСТАНОВКА**

**ЗАДАЧИ**

**1.1 Описание предметной области**

В современном мире музыка играет важную роль в жизни людей. Цифровизация и развитие технологий сделали музыку доступной практически для каждого, вне зависимости от географического положения или социального статуса. Сегодня миллионы слушателей ежедневно используют музыкальные сервисы для прослушивания треков, а авторы песен ищут платформы для распространения своего творчества, взаимодействия с аудиторией и получения обратной связи.

Существующие музыкальные сервисы часто ориентированы только на одну категорию пользователей: либо на слушателей, предлагая им возможность поиска и прослушивания музыки, либо на авторов, предоставляя инструменты для загрузки и распространения контента. Однако интеграция этих двух ролей в едином программном решении открывает новые перспективы, такие как более тесное взаимодействие между авторами и слушателями, создание платформы для обсуждений, оценки и улучшения музыкального материала.

Рынок музыкальных приложений насыщен такими популярными платформами, как Spotify, Apple Music, SoundCloud и другими. Но большинство из них ставят акцент на готовых музыкальных композициях и не предоставляют функционала для начинающих авторов или инструментов, способствующих творческому процессу создания музыки. Таким образом, существует необходимость разработки музыкального сервиса, который будет одновременно удобным инструментом для авторов песен и интерактивной площадкой для слушателей.

**1.2 Постановка задачи**

Целью курсовой работы является разработка программного средства, реализующего музыкальный сервис, который объединяет авторов песен и слушателей на одной платформе.

Для достижения этой цели ставятся следующие задачи:

1 Анализ потребностей пользователей:

– Определить ключевые функциональные требования авторов песен, такие как загрузка треков, создание текстов, возможность получения обратной связи.

– Выявить ожидания слушателей, включая удобство поиска музыки, прослушивания, оценки и комментирования треков.

2 Реализация двух ролей пользователей:

– Авторы песен: возможность публикации своих произведений, взаимодействия с аудиторией, получения статистики прослушиваний и обратной связи.

– Слушатели: возможность искать, прослушивать, оценивать и комментировать музыкальные треки.

3 Создание платформы взаимодействия:

– Реализовать механизм, который позволит авторам получать полезную информацию от слушателей (например, рейтинги или комментарии).

– Внедрить систему рекомендаций, способствующую продвижению авторов и повышению пользовательского опыта для слушателей.

4 Обеспечение удобного интерфейса:

– Разработать интуитивно понятный и привлекательный пользовательский интерфейс, который будет одинаково удобен как для авторов, так и для слушателей.

5 Интеграция современных технологий:

– Использовать базы данных для хранения музыкальных треков и пользовательских данных.

– Реализовать алгоритмы рекомендаций и статистики с помощью аналитических инструментов.

**1.3 Актуальность и обоснование разработки**

Разработка музыкального сервиса, ориентированного одновременно на авторов и слушателей, позволяет не только удовлетворить потребности обеих категорий, но и создать уникальную экосистему, способствующую взаимодействию и взаимной поддержке. Сервис будет полезен:

– Начинающим авторам, которые смогут представить своё творчество широкой аудитории и получить ценную обратную связь.

– Слушателям, которые ищут новые музыкальные произведения и хотят влиять на их развитие, участвуя в обсуждениях и оценке.

Внедрение такого программного средства решает проблему недостаточного внимания к начинающим авторам в крупных музыкальных сервисах и обеспечивает площадку для их роста и популяризации.

Таким образом, разрабатываемое программное средство создаёт новые возможности для обеих сторон и способствует развитию музыкального творчества в целом.

2 ОБОСНОВАНИЕ ВЫБОРА СУБД POSTGRESQL, ОПИСАНИЕ

ВОЗМОЖНОСТЕЙ СУБД POSTGRESQL

Система управления базами данных (СУБД) – совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования баз данных многими пользователями. СУБД играют ключевую роль в современных информационных системах, обеспечивая надежное и удобное управление данными. Среди различных типов СУБД наиболее широко распространены реляционные базы данных. Они основаны на принципе реляционной модели данных, где информация представлена в виде таблиц, состоящих из строк и столбцов. Реляционные СУБД эффективно обрабатывают запросы с использованием языка структурированных запросов SQL.

Известные реляционные СУБД, такие как Oracle, MySQL, Microsoft SQL Server и PostgreSQL, являются востребованными системами управления базами данных. Каждая из этих СУБД имеет свои особенности и преимущества, которые определяют их популярность среди разработчиков и предприятий. Их достоинства заключаются в простой структуре, использовании широко распространённого SQL, быстром обновлении данных.

Для разработки программного средства, реализующего музыкальный сервис для авторов песен и слушателей, была выбрана реляционная СУБД PostgreSQL и графическая оболочка pgAdmin4, позволяющая создавать базы данных и таблицы с помощью встроенного конструктора с возможностью дальнейшего редактирования SQL-скриптов.

**2.1 Функциональные аспекты PostgreSQL как СУБД**

PostgreSQL – это объектно-реляционная система управления базами данных (ORDBMS), наиболее развитая из открытых СУБД в мире. Имеет открытый исходный код и является альтернативой коммерческим базам данных.

Истоки создания PostgreSQL ведут нас в 1986, к университету в Беркли. Профессор Майкл Стоунбрейкер вместе со своими студентами собрал команду для создания новой СУБД, к разработке которой привлекли сотрудников другого проекта этого же учреждения – Ingres. Наработки СУБД претерпели ряд усовершенствований и дополнений, язык POSTQUEL был заменен на SQL [2].

В основе работы СУБД PostgreSQL лежит серверный процесс базы данных, выполняемый на одном сервере. Доступ из приложений к данным базы PostgreSQL производится с помощью специального процесса базы данных. То есть клиентские программы не могут получать самостоятельный доступ к данным даже в том случае, если они функционируют на том же ПК, на котором осуществляется серверный процесс.

PostgreSQL содержит инструменты анализа и оптимизации разбора текста запроса, по аналогии с тем, как это делается в языках программирования. Сначала анализатор (parser) выполняет первоначальный синтаксический и семантический разбор текста запроса. Результатом работы анализатора является дерево разбора. Далее запрос перерабатывается с помощью специальных механизмов обработки запросов (правил). В частности, на этом этапе происходит замена представлений на текст запроса, в результате чего дерево разбора существенно изменяется. Планировщик (planner) выбирает для запроса наиболее подходящий в части минимизация стоимости выполнения план [3].

Некоторые из ключевых функций PostgreSQL включают:

1 Расширенная поддержка SQL: PostgreSQL обеспечивает обширную поддержку стандартного языка SQL, включая сложные запросы, агрегирование данных, оконные функции, подзапросы и многое другое.

2 Хранимые процедуры и функции: PostgreSQL позволяет создавать пользовательские хранимые процедуры и функции на различных языках программирования, таких как PL/pgSQL, PL/Python, PL/Java и другие. Это обеспечивает возможность создания сложной логики обработки данных непосредственно в базе данных.

3 Триггеры: PostgreSQL поддерживает триггеры, которые позволяют автоматизировать выполнение определенных действий при определенных событиях в базе данных.

4 Поддержка JSON и JSONB: PostgreSQL предоставляет встроенную поддержку для работы с JSON и JSONB, что делает его отличным выбором для приложений, использующих нереляционную модель данных.

5 Репликация и высокая доступность: PostgreSQL обладает встроенными механизмами для настройки репликации данных, обеспечивая отказоустойчивость и возможность масштабирования для высокой доступности приложений.

6 Полнотекстовый поиск: С помощью полнотекстового поиска встроенного в PostgreSQL, разработчики могут создавать мощные поисковые функции для обработки текстовых данных.

Таким образом, PostgreSQL предлагает множество ключевых функций, которые делают его мощным инструментом для управления базами данных. Эти функции делают PostgreSQL привлекательным выбором для различных типов приложений и сценариев использования, подчеркивая его роль в сфере управления данными.

**2.2 Обоснование выбора PostgreSQL в качестве СУБД**

Выбор PostgreSQL обосновывается удобством, простотой и широкой функциональностью данной СУБД, а также наличием графической оболочки pgAdmin4.

Возможности СУБД PostgreSQL обширны. Данная СУБД надежна, устойчива, кроссплатформенна. Она использует многоверсионность для обеспечения надежной и быстрой работы в конкурентных условиях под высокой нагрузкой. PostgreSQL отлично использует современную архитектуру многоядерных процессоров. Расширяемость СУБД PostgreSQL поддерживает многочисленные типы данных. Система управления базами данных также позволяет создавать новые типы данных и методы доступа.

Другой важный аспект – масштабируемость. PostgreSQL обладает гибкой архитектурой, которая позволяет легко масштабировать базу данных по мере роста сервиса. Это особенно важно для музыкального сервиса, который может иметь большое количество пользователей и контента.

PostgreSQL также реализует многоверсионное управление параллельным доступом (MVCC), что позволяет нескольким пользователям одновременно работать с базой данных без блокировки доступа. Это ключевое преимущество для музыкального сервиса, где множество пользователей может одновременно осуществлять доступ к базе данных для просмотра, добавления и обновления информации о песнях, альбомах, исполнителях и т.д.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Что такое база данных? [Электронный ресурс]. – Режим доступа : <https://www.oracle.com/cis/database/what-is-database/> – Дата доступа : 27.10.2024.

[2] Что такое PostgreSQL? [Электронный ресурс]. – Режим доступа :  https://selectel.ru/blog/postgresql/ – Дата доступа : 27.10.2024.

[3] Изучаем PostgreSQL [Электронный ресурс]. – Режим доступа :  https://habr.com/ru/companies/otus/articles/706346/ – Дата доступа : 27.10.2024.