

***Автор:***

***Дмитриев Егор Андреевич***

студент

ФГАОУ ВО «Самарский национальный

исследовательский университет

им. академика С.П. Королева»

г. Самара, Самарская область

## **РАЗРАБОТКА БАЗЫ ДАННЫХ ДЛЯ МУЗЫКАЛЬНОГО КАТАЛОГА**

***Аннотация:*** в данной работе рассматривается схема базы данных для приложения «Музыкальный каталог».

***Ключевые слова:*** база данных, музыкальный каталог, приложение.

### ***Введение***

Информационная система «Музыкальный каталог» предназначена для обеспечения удобной навигации пользователей в мире музыки. Обычным пользователям система предоставляет следующие возможности: регистрация на сервисе; поиск музыки по исполнителям, альбомам, названиям песен; оценивание песен; просмотр статистики оценок; составление рекомендаций на основании прослушанных песен. Администраторам система предоставляет дополнительные возможности: добавление, редактирование и удаление исполнителей, альбомов и конкретных песен.

Структура базы данных

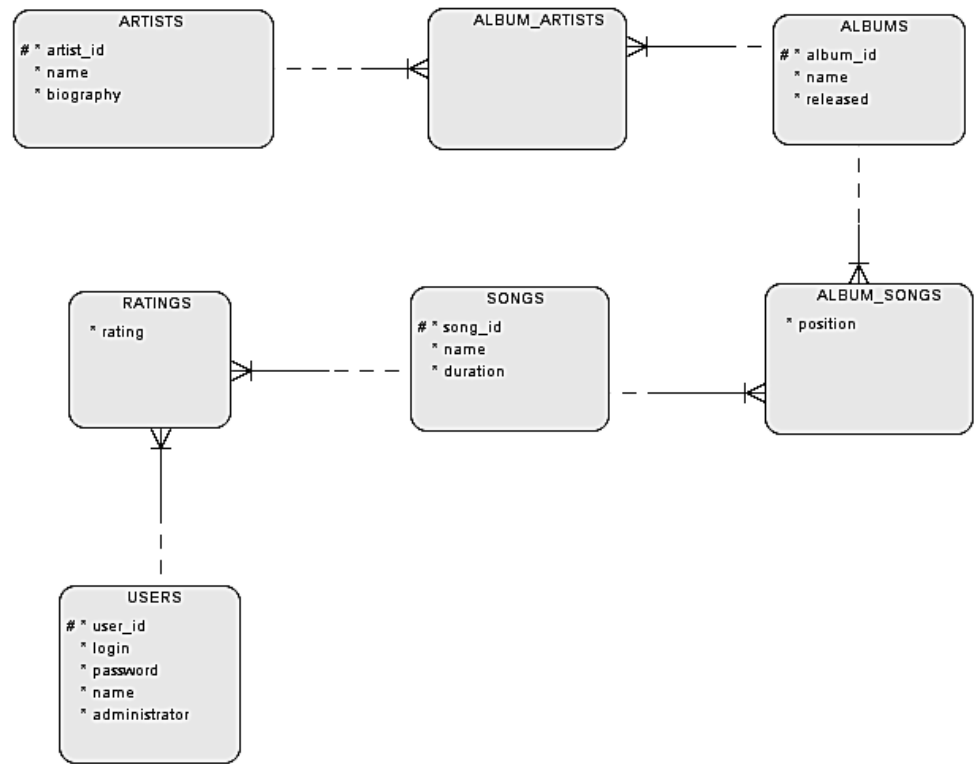


Рис. 1. Логическая схема базы данных

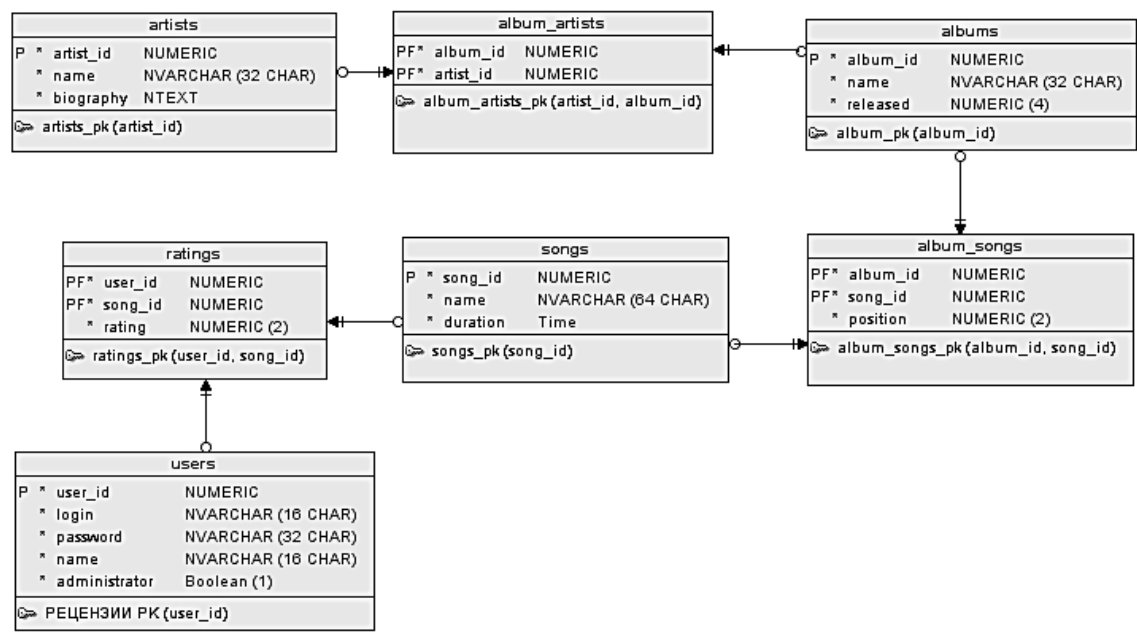


Рис. 2. Физическая схема базы данных

## Описание таблиц

Таблица 1

## Описание таблиц базы данных

Таблица	Атрибут	Описание
artists		Содержит информацию об исполнителях альбомов.
	artist_id	Уникальный идентификатор исполнителя.
	name	Имя исполнителя.
	biography	Краткое описание исполнителя.
albums		Содержит информацию об альбомах.
	album_id	Уникальный идентификатор альбома.
	name	Название альбома.
	released	Год выпуска альбома.
songs		Содержит информацию о конкретной песне.
	song_id	Уникальный идентификатор песни.
	name	Название песни.
	duration	Продолжительность песни.
users		Содержит информацию о зарегистрированных в системе пользователях.
	user_id	Уникальный идентификатор пользователя.
	login	Логин пользователя.
	password	Пароль пользователя. Хранится в хешированном виде.
	name	Имя пользователя.
	administrator	Поле, определяющее, является ли пользователь администратором системы.
album_artists		Содержит информацию об авторах альбомов. Наличие записи в данной таблице означает, что исполнитель является автором указанного альбома (связь многие-ко-многим).
	album_id	Уникальный идентификатор альбома.
	artist_id	Уникальный идентификатор исполнителя.
albums_songs		Содержит информацию о составе альбома. Наличие записи в данной таблице означает, что песня входит в указанный альбом (связь многие-ко-многим).
	album_id	Уникальный идентификатор альбома.
	song_id	Уникальный идентификатор песни.
	position	Позиция песни в альбоме.
ratings		Содержит информацию об оценках пользователей.
	user_id	Уникальный идентификатор пользователя.
	song_id	Уникальный идентификатор песни.
	rating	Оценка пользователя (от 1 до 10).

## *Архитектура приложения*

При написании информационной системы были использованы следующие технологии:

1. MySQL Community Server – свободно распространяемый сервер баз данных.
2. Java Persistence API – средство, предоставляющее возможность сохранять в удобном виде Java-объекты в базе данных.
3. Enterprise Java Beans – компоненты серверного уровня, инкапсулирующие бизнес-логику приложения.
4. Java Server Pages – технология, позволяющая веб-разработчикам легко создавать содержимое, которое имеет как статические, так и динамические компоненты. Наша информационная система также использует библиотеку JSTL, расширяющую спецификацию JSP.
5. На уровне представления используются стандартные технологии: XHTML, CSS, JavaScript.

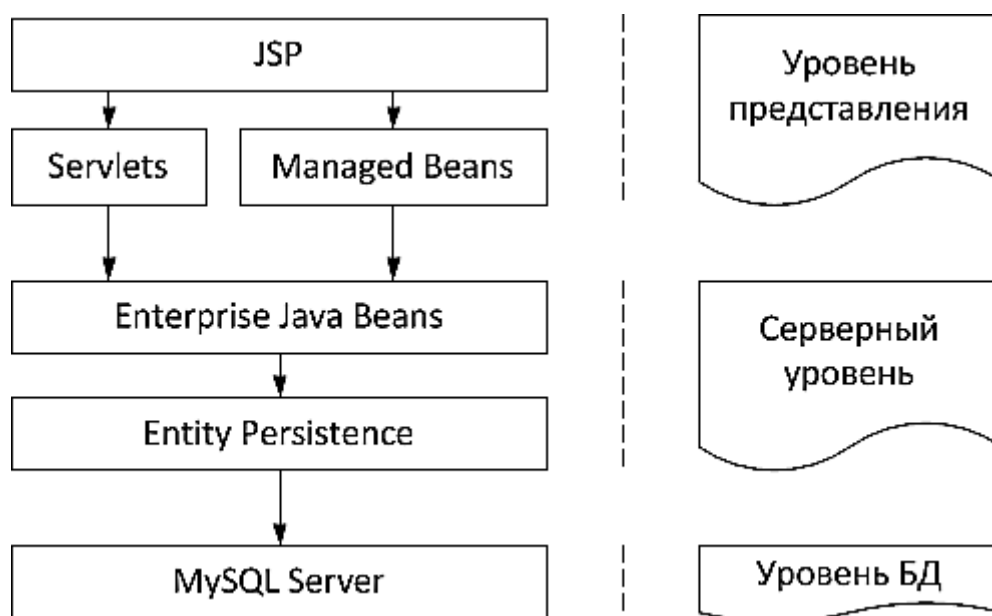


Рис. 3. Общая архитектура приложения

## ***Список литературы***

1. Дейт К.Дж. Введение в системы баз данных. – 8-е изд. – М.: Вильямс, 2005. – 1328 с.

2. Еникеев Р.М. Музыкальный каталог / Р.М. Еникеев, А.С. Адушкин [Электронный ресурс]. – Режим доступа: [https://knowledge.allbest.ru/programming/2c0b65635b3bd69b5c53a88521216c37\\_0.html](https://knowledge.allbest.ru/programming/2c0b65635b3bd69b5c53a88521216c37_0.html) (дата обращения: 24.01.2018).