

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ АКТУАЛИЗАЦИИ МУЗЫКАЛЬНОЙ БИБЛИОТЕКИ НА РАЗЛИЧНЫХ УСТРОЙСТВАХ

Набережнев К.Е., студент гр. ИТм-151, I курс
Научный руководитель: Ванеев О.Н., к.н., доцент
Кузбасский государственный технический университет
имени Т.Ф. Горбачева
г. Кемерово

При работе музыкальными файлами пользователь может столкнуться с проблемой актуализации музыкальной библиотеки на различных устройствах. Актуализация музыкальной библиотеки заключается в обновлении информации о наличных треках, их характеристик на различных устройствах, воспроизводящих музыкальные файлы. На данный момент этот процесс происходит с помощью открытия общих папок и передачи файлов вручную по SMB-протоколу. Этот способ наиболее удобный, но он не использует метаданные. Также часто используемым способом является передача файлов на мобильное устройство через USB, но скорость может варьироваться в зависимости от версий USB. Данный способ также не работает с метаданными.

Метаданными музыкального файла называются данные, которые содержат описательную информацию о музыкальном файле, такую как исполнитель, альбом, название песни, номер трека в альбоме, обложка альбома, и прочую информацию для идентификации файла. Такие метаданные также называют музыкальными тегами. Они заполняются в преобладающем большинстве музыкальных файлов, и это позволяет использовать метаданные для структуризации музыкальной библиотеки.

Цель данной работы состоит в обеспечении наиболее быстрой и удобной актуализации музыкальной библиотеки на различных устройствах.

К числу основных задач работы относятся:

- обеспечение работы с выделением музыкальных файлов;

- возможность копирования музыкальных файлов с использованием заранее заданных шаблонов для создания необходимых папок и переименования файла;
- контроль за дубликатами музыкальных файлов;
- обеспечение кроссплатформенности.

Информационная система разрабатывается в среде Visual Studio 2015 на языке C#. Для обеспечения кроссплатформенности все библиотеки для бизнес-логики пишутся с использованием PCL (Portable Class Library). Это позволит использовать один код для всех платформ, что ускорит разработку и последующую модификацию. Для написания кода, который специфичен на каждой платформе, но используется в портативных библиотеках классов, будет использоваться хитрость, которая называется Bait and Switch. Она позволит портативным библиотекам работать с портативными библиотеками, которые на самом деле не будут использоваться конкретными платформами – конкретная платформа будет использовать свою конкретную реализацию данной портативной библиотеки.

Для обеспечения кроссплатформенности будут использованы технологии UWP (Universal Windows Platform), Xamarin.Forms и WPF (Windows Presentation Foundation). Технология UWP позволит разработать приложение на Windows 10 и Windows 10 Mobile, Xamarin.Forms – на Android, iOS и Windows Phone 8/8.1, а WPF – для более ранних версий Windows.

Для выполнения задач необходима база данных. Хранение информации о музыкальных файлах реализовано с помощью компактной реляционной базы данных SQLite. Заполнение данных происходит автоматически с помощью приложения. Для наиболее удобной разработки приложение использует будет использовать объектно-реляционный модуль сопоставления, позволяющий работать с реляционными данными с помощью объектов.

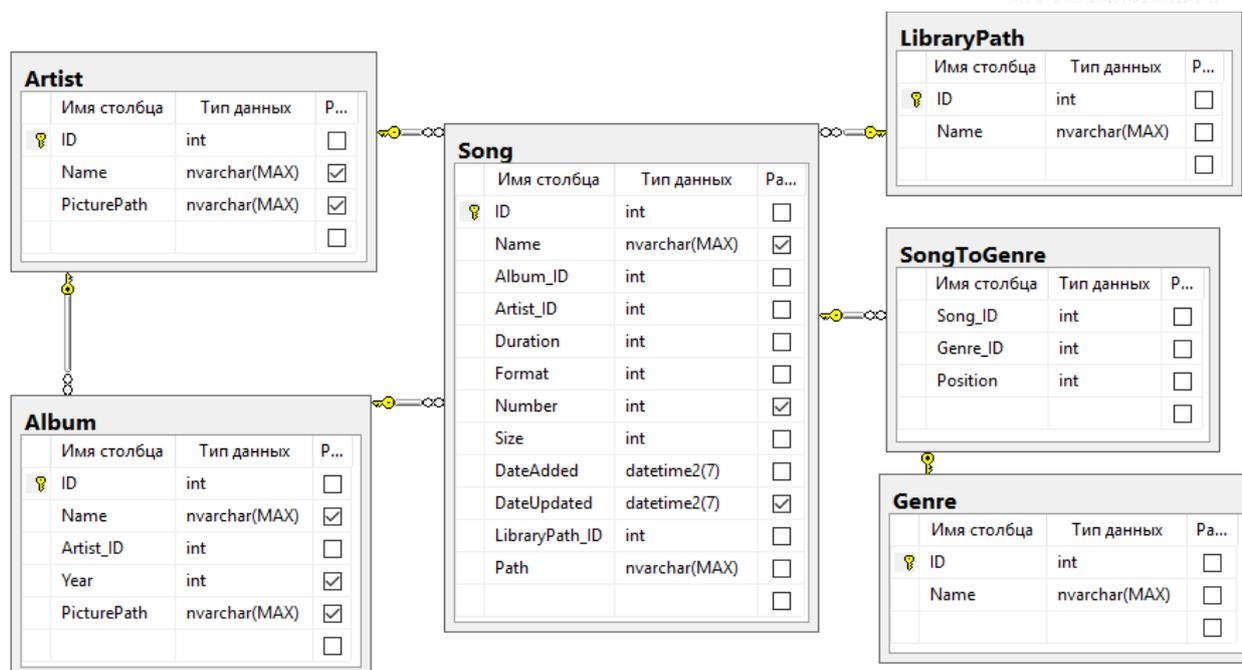


Рисунок 1. ER-диаграмма

База данных будет хранить наиболее важной информации, которая нужна для поиска музыкальных файлов, а также для поддержки функционирования шаблонов для перемещения и контроля за дубликатами: исполнитель, название альбома, год выпуска альбома, количество песен в альбоме, исполнитель альбома (обычно используется для группировки песен разных исполнителей в пределах одного альбома), название песни, номер песни в альбоме, жанр песни, битрейт и формат песни (необходимо для сравнения файлов с одинаковыми тегами), путь до файла. Диаграмма базы данных представлена на рисунке 1. Хранение этих данных позволит обращаться к интересующей информации без дополнительного чтения файлов, что существенно ускорит процесс работы с музыкальными тегами. Обновление музыкальной библиотеки производится либо вручную, либо при добавлении, изменении, удалении музыкальных источников (папок, в которых будет осуществляться поиск музыкальных файлов).

Приложение будет предоставлять удобный поиск для последующего выделения песен, альбомов, исполнителей. Над тем, что было выделено можно выполнять различные операции. Например, переместить согласно шаблонам, применить шаблоны для текста, удалить или скопировать на

другое устройство. Диаграмма последовательности для операции «Копирование выделенных файлов» представлена в приложенном файле.

Возможность перемещения и копирования файлов согласно шаблону с созданием необходимых папок и переименованием файла – одна из ключевых задач данной работы. Пользователь может задать эти шаблоны с использованием информации из тегов. После создания необходимых шаблонов при операциях перемещения или копирования файлов будет произведена соответствующая операция с использованием шаблонов.

Контроль за дубликатами музыкальных файлов производится с помощью сравнения нового музыкального файла, не содержащего данных в базе данных, и старого, содержащего данные в базе данных. Сравнивается показатель качества файла (битрейт и формат), и производятся необходимые операции: удаления файла с худшим качеством и занесение данных о файле с лучшим качеством в базу. Порядок качества пользователь может установить по своему желанию.

Из всего вышесказанного можно сделать следующий вывод: разработанная информационная система позволит решить проблему недостаточно быстрой и удобной работы по актуализации музыкальных библиотек путем хранения в реляционной базе данных музыкальных тегов файлов, обеспечит удобную работу с помощью использования музыкальной базы данных для выбора необходимых песен, а также позволит единым образом организовать локальную музыкальную библиотеку на различных устройствах с помощью копирования.

Список литературы:

1. Microsoft, Entity Framework [Электронный ресурс] – Электрон. текстовые дан. – 2015. – Режим доступа: <https://msdn.microsoft.com/ru-ru/data/ef.aspx>.
2. The Bait and Switch PCL Trick [Электронный ресурс] – Электрон. текстовые дан. – 2014. – Режим доступа: <http://log.paulbetts.org/the-bait-and-switch-pcl-trick/>.

3. Xamarin.Forms [Электронный ресурс] – Электрон. текстовые дан. – 2016. – Режим доступа: <https://developer.xamarin.com/guides/xamarin-forms/>.
4. Введение в универсальную платформу Windows [Электронный ресурс] – Электрон. текстовые дан. – 2016. – Режим доступа: <https://msdn.microsoft.com/ru-ru/library/windows/apps/dn958439.aspx>.