Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра Програмної інженерії

ПОЯСНЮВАЛЬНА ЗАПИСКА ДО КУРСОВОЇ РОБОТИ

з дисципліни “ База даних”

Тема роботи: «ІНФОРМАЦІЙНА СИСТЕМА “КІНОФІЛЬМИ”»

Керівник, доц., к.т.н. каф. ПІ Мазурова О. О.

Студентка гр. ПІ-15-2 Пітюкова М. О.

Комісія:

доц., к.т.н. Мазурова О. О.

ст. викл. каф. ПІ Черепанова Ю. Ю.

ст. викл. каф. ПІ Широкопетлєва М. С.

Харків 2016

Харківський національний університет радіоелектроніки

**Кафедра** \_\_\_Програмної інженерії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Дисципліна** \_\_\_Бази даних\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Спеціальність** \_Програмна інженерія\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Курс** \_\_\_2\_\_\_\_\_\_\_\_\_\_\_\_\_ **Група** \_\_\_ПІ-15-2\_\_\_\_\_\_\_\_\_ **Семестр**\_\_\_\_ 3\_\_\_\_\_\_\_\_\_\_\_

**ЗАВДАННЯ**

**на курсову роботу студента**

\_\_\_\_\_\_\_\_\_\_*Пітюкової Марії Олегівни*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. Тема роботи:** \_\_\_\_«Інформаційна система «*Кінофільми* »\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2. Строк здачі закінченої роботи** \_\_\_23.12.2016\_\_\_\_\_\_\_\_

**3. Вихідні дані для роботи:** методичні вказівки до виконання курсової роботи,\_вимоги до інформаційної системи, предметна область, що пов’язана *з\_кіноматографом* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**4. Зміст розрахунково - пояснювальної записки:** вступ, аналіз предметної області; постановка задачі; проектування бази даних; опис програми; висновки; перелік посилань.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**5. Перелік графічного матеріалу:** загальна схема концептуальної моделі, ER-діаграма, структура 1НФ, 2НФ, 3НФ, схема БД в 3НФ, UML-діаграми, копії екранів (“скріншоти”) прикладної програми, приклади звітів прикладної програми\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**6. Дата видачі завдання** \_\_\_09.09.16 р.\_\_\_

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| Номер | Назва етапів курсової роботи | Строк виконання етапів роботи | Примітки |
| 1 | Аналіз предметної області | 9.09.16 – 24.09.16 | Виконано |
| 2 | Постановка задачі | 20.09.16 – 30.09.16 | Виконано |
| 3 | Побудова ER-діаграми бази даних | 27.09.16 – 15.10.16 | Виконано |
| 4 | Оформлення розділів 1, 2 та 3.1, 3.2 пояснювальної записки | 15.10.16 - 27.10.16 | Виканано |
| 5 | Перша контрольна точка з курсового проекту | 24.10.16 – 28.10.16 | Виконано |
| 6 | Нормалізація бази даних | 20.10.16 - 10.11.16 | Виканано |
| 7 | Створення демо-версії програми | 20.10.16 – 20.11.16 | Виканано |
| 8 | Тестування програми, наповнення бази даних | 15.11.16 - 25.11.16 | Виканано |
| 9 | Друга контрольна точка з курсового проекту | 21.12.16– 02.12.16 | Виканано |
| 10 | Реалізація остаточної версії програми | 1.12.16-15.12.16 | Виканано |
| 11 | Оформлення інших розділів пояснювальної записки | 1.11.16 – 15.12.16 | Виканано |
| 12 | Захист курсового проекту (третя контрольна точка) | 12.12.16- 23.12.16 | Виканано |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Пітюкова М.О.

Керівник \_\_\_\_ доц. Мазурова О.О.

« 09 » вересня 2016 р.

РЕФЕРАТ / ABSTRACT

Пояснительная записка к курсовой работе: 37 страниц, 4 раздела, 15 подразделов, 28 рисунков, 5 источников.

Целью курсовой работы является создание информационной системы, которая облегчит процессы хранения и использования информации, для студентов, которые изучают киноискусство или увлекаются.

Средствами и методами разработки являются Visual Studio, язык SQL, язык программирования C#.

В результате спроектировали базу данных и создали информационную систему «Кинофильмы», в виде десктопного приложения для работы с данными. Часть приложения, содержащая основной функционал, имеет возможность хранить, редактировать, изменять и удалять различные данные о разделах фильмов, актеров, режиссёров. Пользователь имеет возможность сохранять или распечатывать информацию о фильме в формате pdf, doc, а также результаты произвольного запроса к базе.

БАЗА ДАННЫХ, СУБД, ИНОФРМАЦИОННАЯ СИСТЕМА, КИНОФИЛЬМ, ЯЗЫК ПРОГРАММИРОВАНИЯ C#, MSSQL, VISUAL STUDIO, ПРОГРАММА.

СОДЕРЖАНИЕ

[Введение](#page6) [6](#page6)

[1 Анализ и концептуальное моделирование предметной области](#page7) [7](#page7)

[1.1 Анализ предметной области](#page7) [7](#page7)

1.2 [Описание функциональной структуры системы](#page8) [8](#page8)

[1.3 Описание информационных потребностей](#page10) [10](#page10)

[1.4 Описание документооборота системы](#page11) [11](#page11)

[1.5 Описание алгоритмических зависимостей](#page11) [11](#page11)

[1.6 Описание ограничений целостности](#page12) [12](#page12)

[2 Постановка задачи](#page13) [13](#page13)

[3 Проектирование базы данных](#page15) [15](#page15)

[3.1 UML-моделирование](#page15) [15](#page15)

[3.2 Построение ER-диаграммы](#page16) [16](#page16)

[3.2 Нормализация схемы базы данных](#page18) [18](#page18)

[4 Описание программы](#page22) [22](#page22)

[4.1 Общие сведения](#page22) [22](#page22)

4.2 Запуск программы [22](#page22)

[4.3 Назначение и логическая структура](#page23) [23](#page23)

[4.4 Описание физической модели данных](#page24) [24](#page24)

[4.5 Описание программной реализации](#page26) [26](#page26)

[4.6 Описание задачи автоматизации](#page39) [39](#page39)

[Выводы](#page42) [42](#page42)

[Перечень ссылок](#page43) [44](#page43)

ВВЕДЕНИЕ

С давних времен проблема хранения и структурирования информации являлась актуальной. Таковой она остается и на сегодняшний день. По мере развития технологий были испробованы различные способы хранения информации. Сегодня самым распространённым из них являются различные базы данных.

Существует множество систем, нуждающихся в использовании баз данных для более эффективной работы с большим объемом информации. Ярким примером являются различные интернет-справочники.

Не смотря на бурное развитие технологий, которые появляются, развиваются и устаревают так быстро, что трудно не только изучить их все, а даже уследить за ними, у многих до сих пор не угасает интерес к такому старому, но в тоже время и современному искусству, как кинематограф.

Существует много жанров данного направления: детективы, комедии, боевики, фантастика, приключения… Создано множество кинокартин, и из-за чего выбирать, что посмотреть становиться труднее. И здесь на помощь может прийти информационная система, реализованная в виде десктопного приложения – информационная система «Кинофильмы».

Во время курсового проектирования была разработана вышеуказанная система, позволяющая получить доступ к информации о жанрах фильма, его режиссерах, актерах. Также предусмотрена возможность добавления, удаления и редактирования информации в базе данных. Система также способна генерировать отчеты – информацию о конкретном фильме и результаты произвольного запроса к базе. Приложение обладает понятным, гибким интерфейсом, который позволяет пользователям легко и быстро ориентироваться.

1 АНАЛИЗ И КОНЦЕПТУАЛЬНОЕ МОДЕЛИРОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

* 1. Анализ предметной области

Киноискусство, род искусства, произведения которого создаются с помощью киносъёмки реальных, специально инсценированных или воссозданных средствами мультипликации событий действительности. В киноискусстве синтезируются эстетические свойства литературы, театрального и изобразительного искусств, музыки на основе собственно лишь ему присущих, выразительных средств, из которых главными являются фотографическая природа изображения, позволяющая с предельной достоверностью воссоздавать любые картины действительности, и монтаж.

Общее понятие кинематографа имеет более конкретные разделы в зависимости от школы разных стран, например в Америке явными представителями канона сего искусства является Квентин Тарантино, Джим Джармуш, Дэвид Линч. В кинокартинах затрагиваются различные человеческие проблемы, чувства. Фильмы неоднотипны, они делятся на жанры: детективы, фантастика, мультфильмы и др.

Сведения о фильмах, режиссерах, актерах систематизируют в виде автобиографий, энциклопедий, журналов, однако это довольно затратно и взаимодействовать с ними достаточно долго и не всегда просто. Лучшая альтернатива – информационная система, представленная в виде десктопного приложения в образовательных целях. Она доступна для всех, у кого есть компьютер и подключение к сети Интернет, а работать с ней быстрее и удобнее при наличии возможности поиска, а также фильтрации, сортировки по таким параметрам, как название, год, страна, жанр, киностудия, рейтинг. Пользователь сможет получить подробное описание кинокартины с учетом вышеуказанных параметров. При реализации таких возможностей не обойтись без базы данных, которая и будет хранить всю информацию о фильмах. Также должна быть предусмотрена возможность редактирования базы кинофильмов специально уполномоченным лицом – администратором.

* 1. Описание функциональной структуры системы

На рисунке 1.1 представлено описание функциональной структуры системы, на ней описаны взаимосвязи основных участников взаимодействия в предметной области.

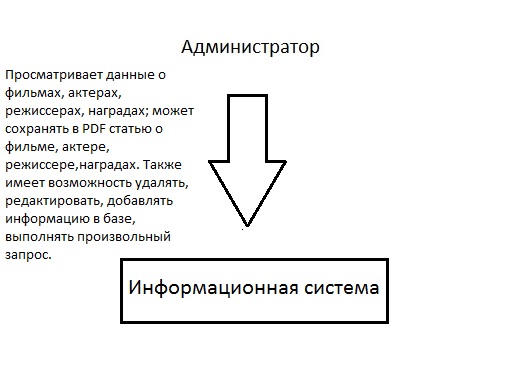


Рисунок 1.1 – Функциональная структура системы

* 1. Описание объектов, атрибутов и их связей

На основе анализа предметной области можно выделить следующие объекты:

1. кинокартины;
2. режиссеры;
3. актеры;
4. награды.

На рисунке 1.2 представлено описание объектов предметной области и их взаимосвязей. Данная структура включает в себя режиссера, актеров, награды, фильм.

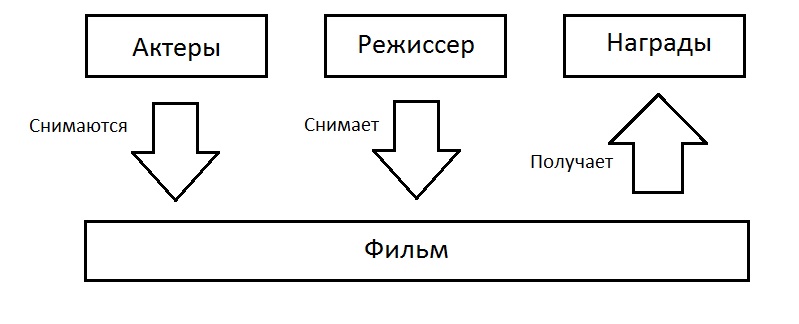


Рисунок 1.2 – Описание объектов и их связей

Соответственно предметной области, база данных должна хранить следующую информацию:

а) данные о фильмах:

1. название;
2. год;
3. страна;
4. киностудия;
5. режиссер;
6. жанр;
7. актеры;
8. время;
9. описание;
10. рейтинг;

11) награды.

б) данные об актерах:

1. фамилия, имя;
2. год рождения;
3. страна;
4. дополнительная информация.

в) данные о режиссерах:

1. фамилия, имя;
2. дата рождения;
3. страна;
4. дополнительная информация.

г) данные о наградах:

1. название награды;
2. номинация;
3. год;
4. победитель.

1.4 Описание информационных потребностей

На основе анализа предметной области выделим основные процессы и функции, которые должна реализовать система:

1. поиск фильмов по названию, жанру, стране, киностудии;
2. поиск актеров по фамилии, стране;
3. поиск режиссеров по фамилии, стране;
4. просмотр информации о каждом фильме;
5. вывод фильмов конкретной категории (фильтрация);
6. возможность сохранения в PDF или DOC страницы с информацией о

фильмах, актерах, режиссерах;

1. добавление информации о фильме, актере, режиссере;
2. редактирование/удаление данных базы с поиском по названию

фильма, актере, режиссере;

1. возможность обращения к базе произвольным SQL-запросом с

дальнейшим выводом результата и возможностью его печати;

1. сортировка фильмов, актеров, режиссеров по всем полям;
2. фильтрация фильмов по годам, жанру, рейтингу;
3. вход/выход в систему;
4. считывание данных о фильмах с текстового документа.

Данная информационная система предназначена для более удобного и наглядного получения сведений о кинокартинах в учебных целях: сокращения количества времени, необходимого для поиска информации в автобиографиях, энциклопедиях, журналах, а также для простого редактирования базы фильмов.

Система предназначена для человека с правами администратора.

Задача автоматизации должна быть реализована в следующем виде. На главной форме реализовать (по клику на поле меню) считывания информации о фильмах (название, год, страна, киностудия, режиссер, жанр, актеры, время, описание, рейтинг, награды) с текстового файла и заполнение базы по соответствующим полям.

В программе реализовано отображение таких статистических данных, как топ 5 рейтинговых фильмов, количество фильмов снятых по годам, количество фильмов определенных жанров.

1.5Описание документооборота системы

Система способна генерировать отчёт, который представляет собой сохраняемую в формате PDF страницу с информацией о фильмах (название, год, страна, киностудия, режиссер, жанр, актеры, время, описание, рейтинг, награды), актерах (имя, фамилия, год рождения, страна, дополнительная информация), или режиссёрах (имя, фамилия, год рождения, страна, дополнительная информация). А также в формате DOC с информацией о фильмах (название, год, режиссер, жанр, описание). Также есть текстовый документ с информацией о фильмах (название, год, страна, киностудия, режиссер, жанр, актеры, время, описание, рейтинг, награды).

1.6 Описание алгоритмических зависимостей

Пока в базе данных находится хотя бы один фильм, в ней останется как минимум один актер и как минимум один режиссер.

После добавления фильма, делается проверка, с помощью которой определяется, есть фильм в базе данных или нет. Проверка проходит по названию, жанру, стране, киностудии.

Когда администратор входит в базу, проходит проверка – есть ли такой логин, после проверяется пароль.

* 1. Описание ограничений целостности

Для нормального функционирования базы данных вводятся следующие ограничения:

1. в одном фильме могут играть несколько актеров, а один актер может

появляться во многих фильмах;

1. несколько фильмов относятся к одному режиссеру;
2. каждый фильм не может иметь больше одного уникального номера;
3. каждый актер не может иметь больше одного уникального номера;
4. каждый режиссер не может иметь больше одного уникального номера;
5. после регистрации никто не может поменять логин администратора;
6. удаляет, редактирует, добавляет данные о фильмах, актерах,

режиссерах в базе администратор.

1. ПОСТАНОВКА ЗАДАЧИ

Исходя из анализа предметной области и результатов концептуального моделирования, можно однозначно решить какие именно функции должна выполнять система.

Информационная система должна предоставлять пользователю такие функции, как:

1. просмотр информации о фильмах, хранящейся в БД;
2. редактирование, добавление, удаление информации о фильмах, актерах, режиссерах;
3. поиск фильмов по названию, жанру, стране, киностудии;
4. поиск актеров по фамилии, стране;
5. поиск режиссеров по фамилии, стране;
6. сортировка фильмов по всем полям;
7. сортировка актеров по всем полям;
8. сортировка режиссеров по всем полям;
9. фильтрация фильмов по годам, жанру, рейтингу;
10. просмотр статистики;
11. занесение данных о фильмах с текстового документа;
12. возможность сохранения в PDF или DOC страницы с информацией о

фильмах, актерах, режиссерах;

1. вход на страницу администратора через форму авторизации с логином и паролем;
2. возможность выхода со страницы администратора;
3. возможность обращения к базе произвольным SQL-запросом с

дальнейшим выводом результата и возможностью его печати.

Также должна быть организована кодировка пароля администратора.

На главном окне программы можно найти отображение статистических данных: 5-ть рейтинговых фильмов, количество фильмов снятых по годам, количество фильмов определенных жанров.

Задачей автоматизации является считывание данных о фильмах (название, год, страна, киностудия, режиссер, жанр, актеры, время, описание, рейтинг, награды) с текстового документа и запись в базу данных.

Приложение должно обладать удобным для пользователя интерфейсом, позволяющим наглядно отображать и помочь быстро воспринимать найденную информацию [1].

1. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

3.1 UML-моделирование

Конечные цели создания информационной системы «Кинофильмы» и функции, которые она будет выполнять, удобно представить на диаграмме Use case [2]. Для этого необходимо выделить границу системы, основного актера, который будет ее использовать и основные функции, которым будет пользоваться актер.

Десктопным приложением в образовательных целях пользуется администратор.

На рисунке 3.1 представлена диаграмма прецедентов (диаграмма Use case) информационной системы «Кинофильмы».

На диаграмме прецедентов отображено взаимодействие администратора с информационной системой. ИС предоставляет разнообразный функционал, позиции которого отображены в вариантах использования.

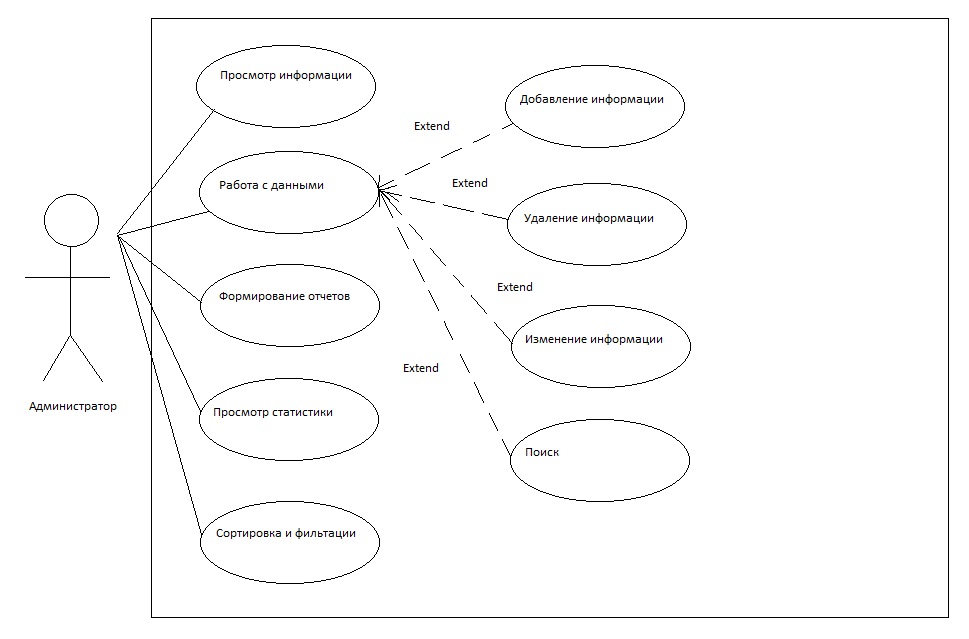


Рисунок 3.1 – Диаграмма Use case «Кинофильмы»

3.2 Построение ER-диаграммы

Исходя из набора объектов и их атрибутов, полученных на этапе концептуального моделирования, была построена ER-диаграмма, которая изображена на рисунке 3.2 [2] и определенны такие сущности:

а) данные о фильме, атрибутами являются номер фильма, название, год,

страна, киностудия, режиссер, жанр, актеры, длительность, описание, рейтинг,

награды (первичный ключ – номер фильма);

б) данные об актере, атрибутами являются номер актера, фамилия и имя,

год рождения, странна, дополнительная информация (первичный ключ – номер актера);

в) данные о режиссере, атрибутами являются номер режиссера, фамилия и

имя, год рождения, странна, дополнительная информация (первичный ключ – номер режиссера);

г) данные о награде, атрибутами являются номер награды, название,

номинация (первичный ключ – номер награды);

д) данные об актерах фильма, атрибутами являются номер фильма, номер

актера (первичный ключ - номер фильма, номер актерам);

е) данные о наградах фильма, атрибутами являются номер фильма, номер

награды, год, победитель;

ж) связь «Один ко многому» между сущностями «Фильм» и «Режиссер»,

так как у фильма может быть один режиссер, а у одного режиссера – несколько фильмов;

з) связь «Многие ко многим» между сущностями «Фильм» и «Актер»,

«Фильм» и «Награда»: у одного фильма может быть несколько актеров/ наград, и один актер/одна награда может играть/быть в/у нескольких фильмах — для нескольких фильмов.

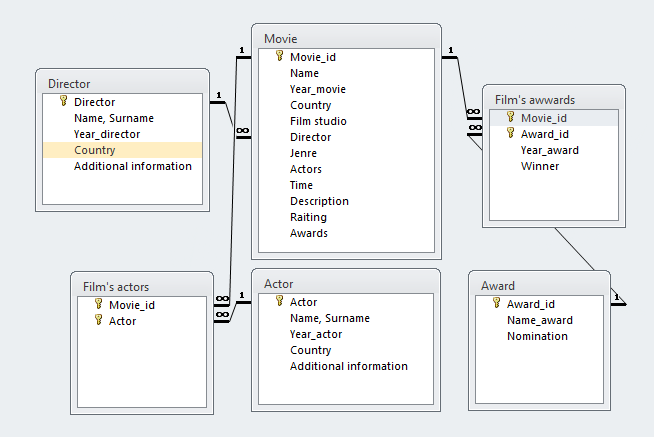


Рисунок 3.2 – ER-диаграмма предметной области «Кинофильмы»

3.3 Нормализация схемы базы данных

Нормализация - разделение таблицы на две или более с целью исключения избытка и избежать аномалий баз данных [3].

Для того чтобы база данных не содержала аномалий, необходимо последовательно привести ее к трем нормальных формам. Таблица находится в первой нормальной форме в случае, если все атрибуты таблицы атомарны. Для того чтобы добиться этого, необходимо найти первичный ключ, через который можно получить данные любого из атрибутов. На рисунке 3.3 изображено отношение Т, также в нем выбраны первичными ключами «Id\_movie», «Id\_award», «Id\_actor» которого достаточно, чтобы иметь доступ ко всем оставшимся полям, таким образом, данная таблица приведена к первой нормальной форме. В первом столбце стрелок отображается наличие зависимости всех полей от ключевого поля «Id\_movie», «Id\_award», «Id\_actor».

Для того чтобы отношение находилось во второй нормальной форме, оно должно находиться в первой нормальной форме и необходимо, чтобы все атрибуты имели полную функциональную зависимость от первичного ключа. На рисунке 3.3 отображены атрибуты и первичные ключи, которые составляют таблицу, находящуюся во второй нормальной форме. Каждый атрибут зависит от первичного ключа и не зависит от других полей, составляющих его, таким образом можно утверждать, что отношение Т находится во второй нормальной форме.

Третья нормальная форма предполагает отсутствие транзитивных связей между неключевыми атрибутами. Если таковы присутствуют, то необходимо производить нормализацию таблицы к третьей нормальной форме путем разделения начальной таблицы на другие посредством отделения атрибутов, участвующих в транзитивных связях.

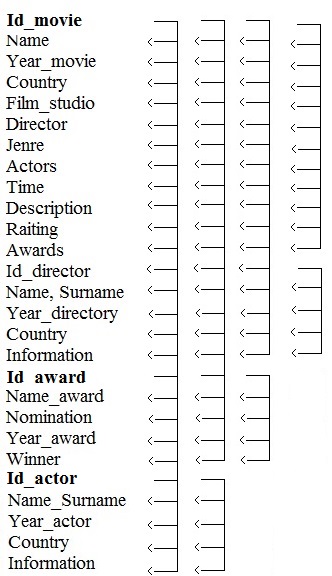


Рисунок 3.3 – Универсальное отношение Т с функциональными зависимостями

Для приведения к третьей нормальной форме необходимо выделять из набора атрибутов на рисунке 3.4 таблицы.

При отделении первой зависимости мы получим таблицу «Actors» (см. рис. 3.4) с первичным ключом «Id\_actor» и атрибутами: «Name, Surname», «Year\_actor», «Country», «Information».

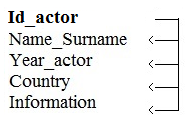


Рисунок 3.4 - таблица «Actors»

При отделении второй зависимости мы получим таблицу «Awards» (см. рис. 3.5) с первичным ключом «Id\_award» и атрибутами: «Name\_award», «Nomination», «Year\_award», «Winner».

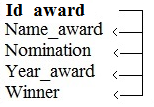


Рисунок 3.5 – таблица «Awards»

Далее мы получим таблицу «Т1»:

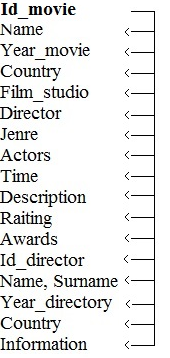


Рисунок 3.6- Таблица «Т1»

При отделении зависимости мы получим таблицу «Director» с первичным ключом «Id\_director» и атрибутами: «Name, Surname», «Year\_directory», «Country», «Information» и таблица «Movies» с первичным ключом «Id\_movie» и атрибутами: «Name», «Year\_movie», «Country», «Film\_studio», «Director», «Jenre», «Actors», «Time», «Description», «Raiting», «Awards».

Результатом нормализации является схема базы данных, которая не содержит аномалий, представленная на рисунке 3.7.

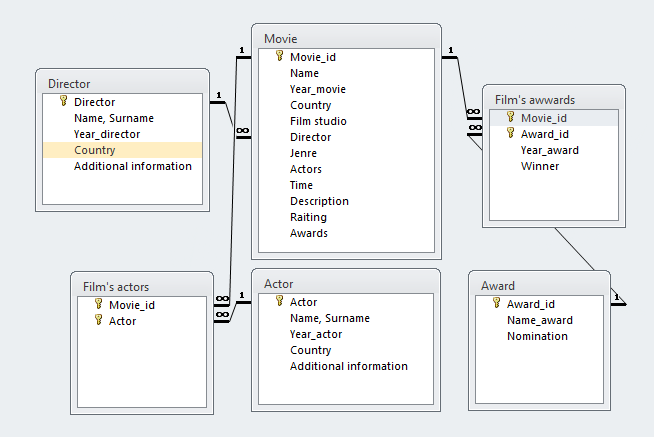


Рисунок 3.7 – Схема базы данных

4 ОПИСАНИЕ ПРОГРАММЫ

4.1 [Общие сведения](#page22)

Разработанный программный продукт был создан с помощью языка программирования C# в среде MS Visual Studio 2015. Также для реализации интерфейса пользователя использовалась технология Windows Forms. В качестве СУБД была выбрана MSSQL 2015, которая представляет собой СУБД от компании Microsoft. При выборе данной системы управления базами данных мной учитывались такие факторы, как гибкость при ее внедрении в прикладное приложение, быстрое взаимодействие, а также удобная интеграция с Visual Studio. Также MSSQL является одной из наиболее часто используемых в программной инженерии СУБД, таким образом ознакомление с ней даст необходимый опыт, который может пригодиться в дальнейшей практике.

Технические требования для работы с программой:

a) операционная система Windows;

b) MSSQL 2015;

c) процессор с тактовой частотой 1.8 ГГц;

d) монитор с разрешением не ниже 1024 x 768.

Программа занимает вместе с базой данных занимают около 25,7 Мб памяти на жестком диске.

4.2 Запуск программы

Для того чтобы запустить программу, нужно открыть папку с программой на локальном диске и запустить исполнительный файл «BDMovies.exe». После этого следует открытие программы.

4.3 Назначение и логическая структура

Всю реализацию функционала программы можно разделить на модули, которые соответствуют решаемым программой задачам.

Первый модуль – модуль отображения базы данных. Он включает в себя удобное представление таблиц базы данных. Также данный модуль включает в себя реализацию функций поиска в таблицах по всем их полям, фильтрацию элементов по некоторым критериям, сортировку таблиц по всем полям, удаление/редактирование/добавление данных в таблицы.

Второй модуль – статистика. Он отвечает за отображение статистической информации. Там производятся определенные расчеты, результаты которых заносятся в отображение и полученные данные отображаются пользователю.

Третий модуль – редактор запросов. Он отвечает за выполнение пользовательских SQL-запросов. На вход функции принимается строка SQL-запроса, а на выходе пользователь получает таблицу или единичное значение, которое является результатом выполнения запроса.

Четвертый модуль – формирование информации в отчеты разного формата. Данный модуль является достаточно обширным, в нем происходит работа с Microsoft Office и Acrobat Reader. Суть работы состоит в том, что пользователь выбирает одну из таблиц, и программа, получив данные, записывает их в шаблон – предварительно сформированный документ - и по окончанию работы открывает результирующий .doc или .pdf файл.

4.4 Описание физической модели базы данных

База данных состоит из шести таблиц, структуры и краткие описания которых представлены ниже.

Таблица «Movies» содержит данные о фильмах. Ниже представлена структура таблицы (см. рис. 4.1)

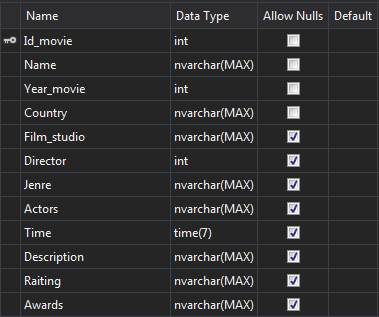


Рисунок 4.1 – Структура таблицы Movies

Таблица «Director» содержит данные о режиссерах. Ниже представлена структура таблицы (см. рис. 4.2)

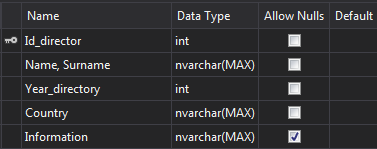


Рисунок 4.2 – Структура таблицы Director

Таблица «Actors» содержит данные об актерах. Ниже представлена структура таблицы (см. рис. 4.3)

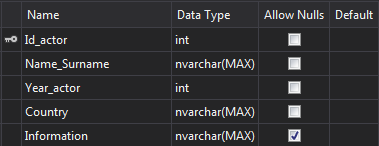


Рисунок 4.3 – Структура таблицы Actors

Таблица «Films actor» содержит данные об актерах фильмов. Ниже представлена структура таблицы (см. рис. 4.4)

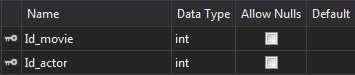


Рисунок 4.4 – Структура таблицы Films actor

Таблица «Awards» содержит данные о наградах. Ниже представлена структура таблицы (см. рис. 4.5)

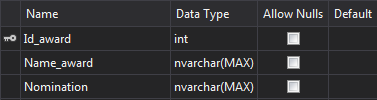


Рисунок 4.5 – Структура таблицы Awards

Таблица «Films awards» содержит данные о наградах фильмов. Ниже представлена структура таблицы (см. рис. 4.6)

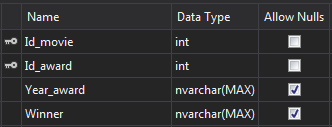


Рисунок 4.6 – Структура таблицы Films awards

4.5 Описание программной реализации

При запуске программы открывается окно входа (см. рис. 4.7).

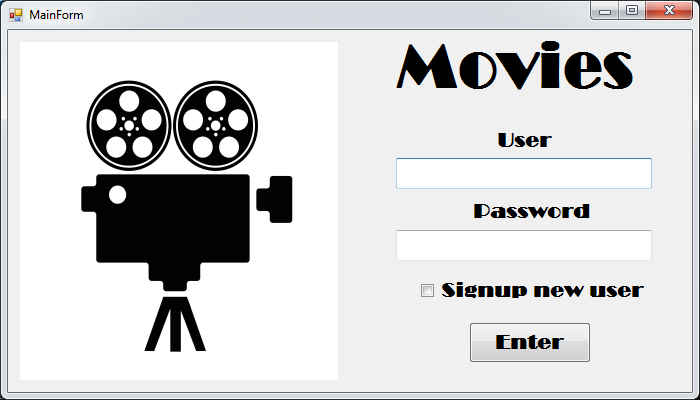


Рисунок 4.7 – Вход в программу

Если будет введен не существующий логин администратора или неправильный пароль и нажата кнопка «Enter», то будет показана ошибка (см. рис. 4.8). Если администратор не зарегистрирован, то следует поставить галочку возле надписи «Singup new user» на окне входа в программу (см. рис. 4.7).

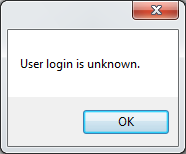
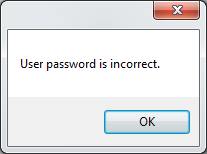
 

Рисунок 4.8 – Ошибки при входе в программу

После закрытия окна входа открывается главное окно программы (см. рис. 4.9). В окне администратор может посмотреть какие фильмы есть в базе данных на данный момент.

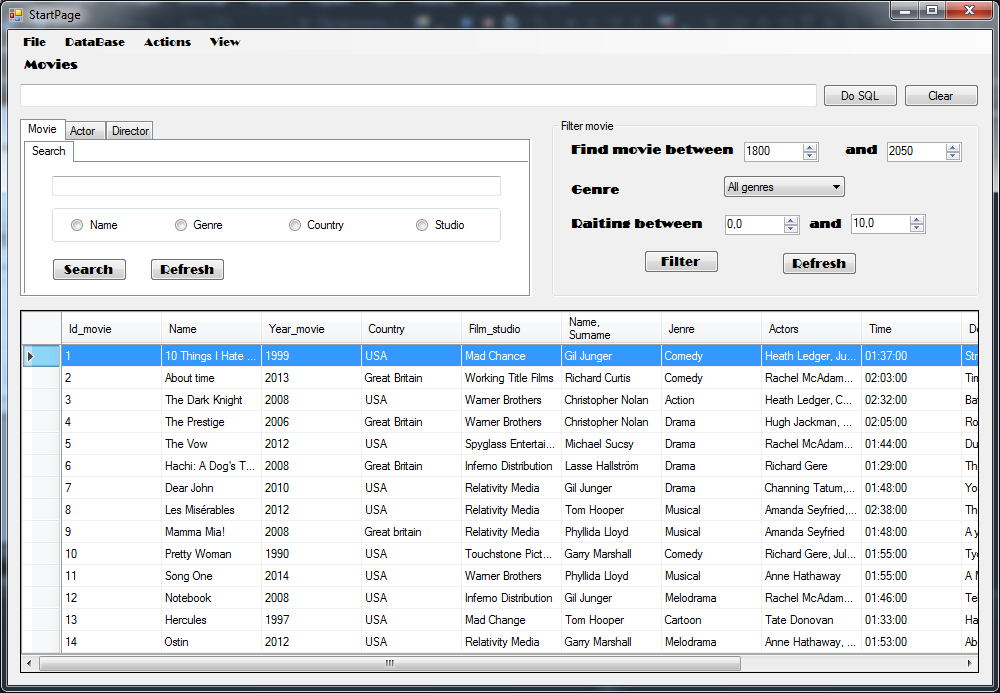


Рисунок 4.9 – Главное окно программы

«File», на панели главного меню, открывает контекстное меню, в котором мы можем выбрать такие функции, как «Open» (открыть документ), «Save» (сохранить таблицы данных в формате \*pdf или \*doc) и «Exit» (выйти из программы). При нажатии на кнопку «DataBase», можно выбрать какую из таблиц базы данных вы хотите отобразить в главном окне. В «Actions» можно выбрать функцию для работы с информацией в таблицах. Например, «Add» - для того, чтобы добавить запись, «Update» - чтобы изменить, «Delete» - чтобы удалить, предварительно выбрав элемент и нужную таблицу. «View» предоставляет возможность просмотра статистики, нажав на «Statistic» в контекстном меню и выбрав нужную.

Ниже приведен пример внешнего вида окна редактирования для таблицы «Movies» (см. рис. 4.10). Как видно на рисунке, информация из строки, которую пользователь хочет изменить, загружается в данное окно.

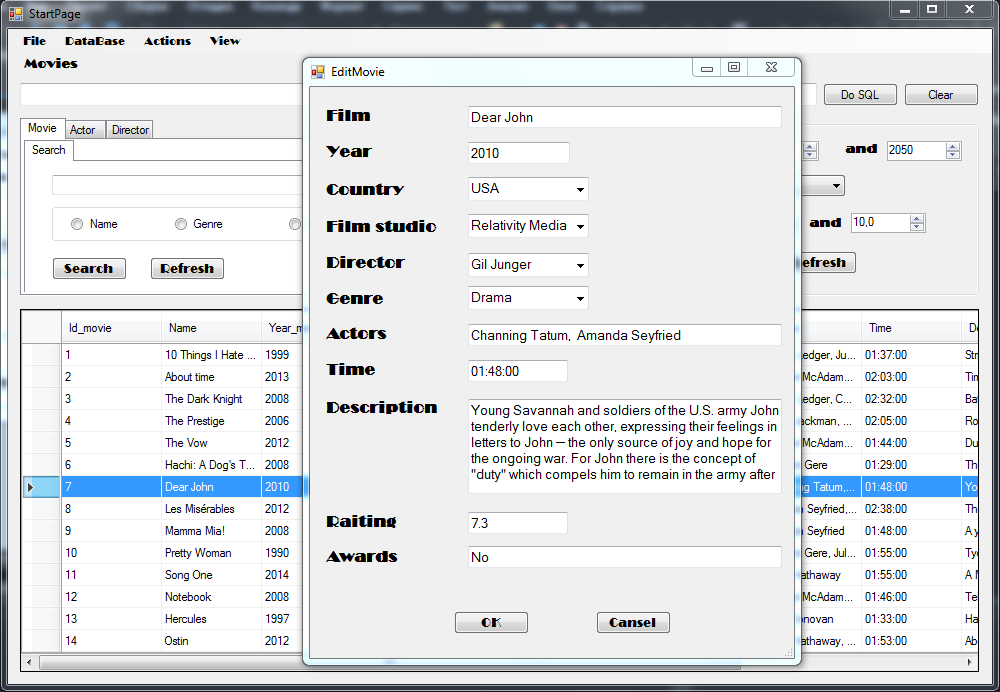


Рисунок 4.10 – Окно редактирования информации о фильме

Внешний вид окна добавления полностью совпадает с внешним видом окна, представленного на рисунке 4.10. Единственное отличие состоит в том, что изначально все поля с данными пусты и пользователю необходимо самому их инициализировать.

Перейдем к вспомогательному функционалу окна отображения таблиц (см. рис. 4.9). Сверху видно два блока, каждый из которых отвечает за свою функцию. Первый блок – поиск. В нем имеется три вкладки. Можно выбрать по какой таблице и по какому из полей делать поиск. Работу функции поиска можно увидеть на рисунке 4.11.

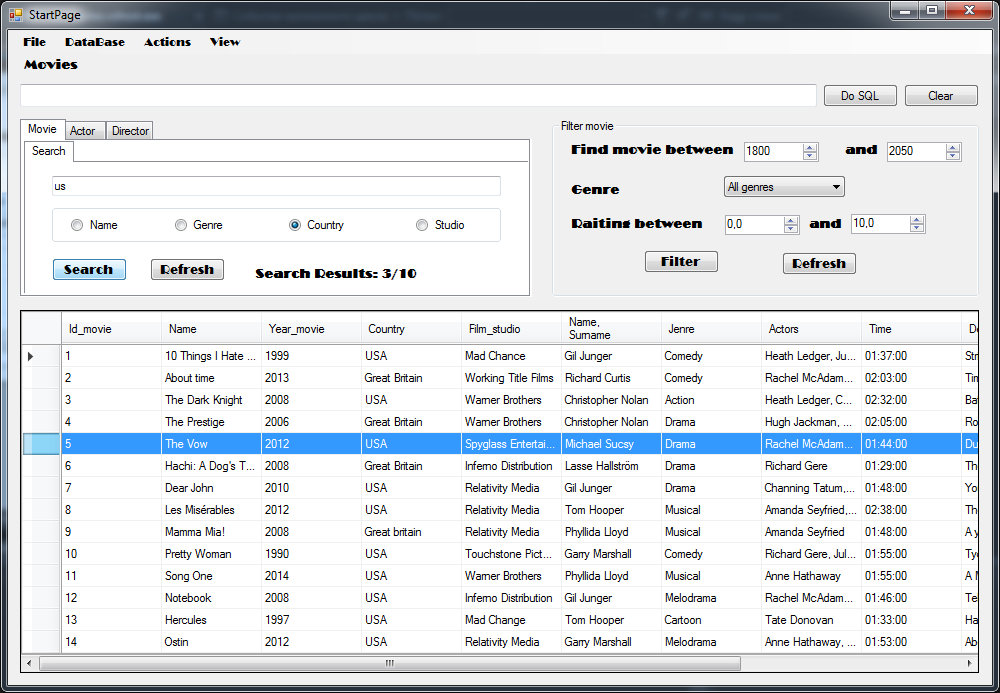


Рисунок 4.11 – Поиск по содержимому таблицы

Следующий блок – фильтрация. Фильтрация записей таблицы «Movies» происходит по таким критериям, как временной промежуток выхода фильма, его жанр и диапазон рейтинга. Работу функции фильтрации можно увидеть на рисунке 4.12.

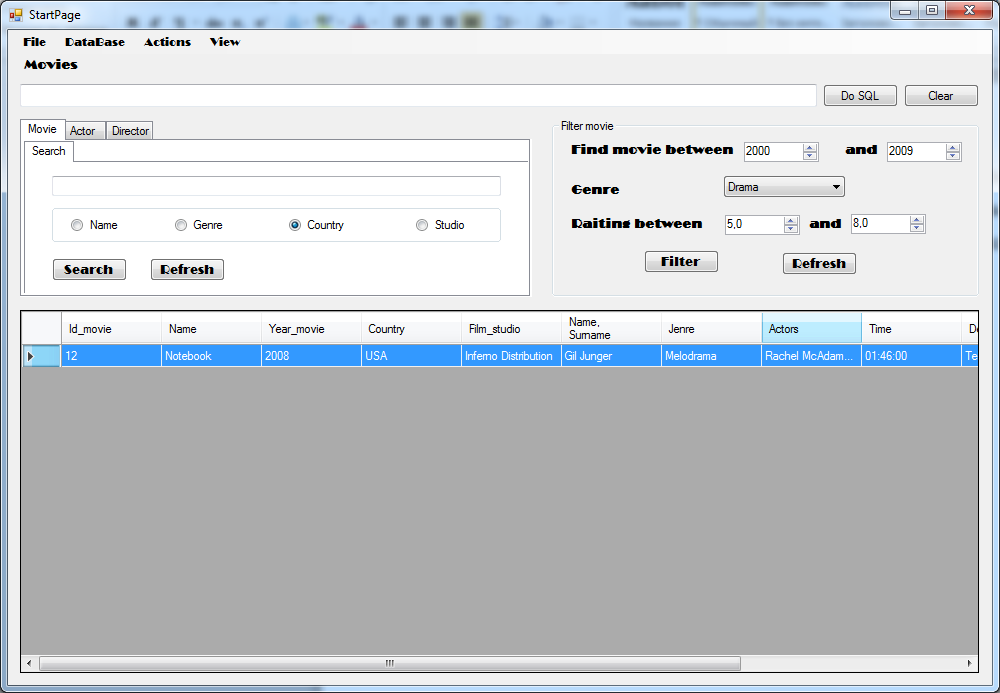


Рисунок 4.12 – Фильтрация по содержимому таблицы

Следующая из возможностей программы – написание собственных SQL-запросов и получение их результата. Введя произвольный запрос в поле textbox главного окна и нажав «Do SQL», мы увидим нужные нам данные в таблице. Кнопкой «Clear» можно очистить поле textbox (см. рис. 4.13).

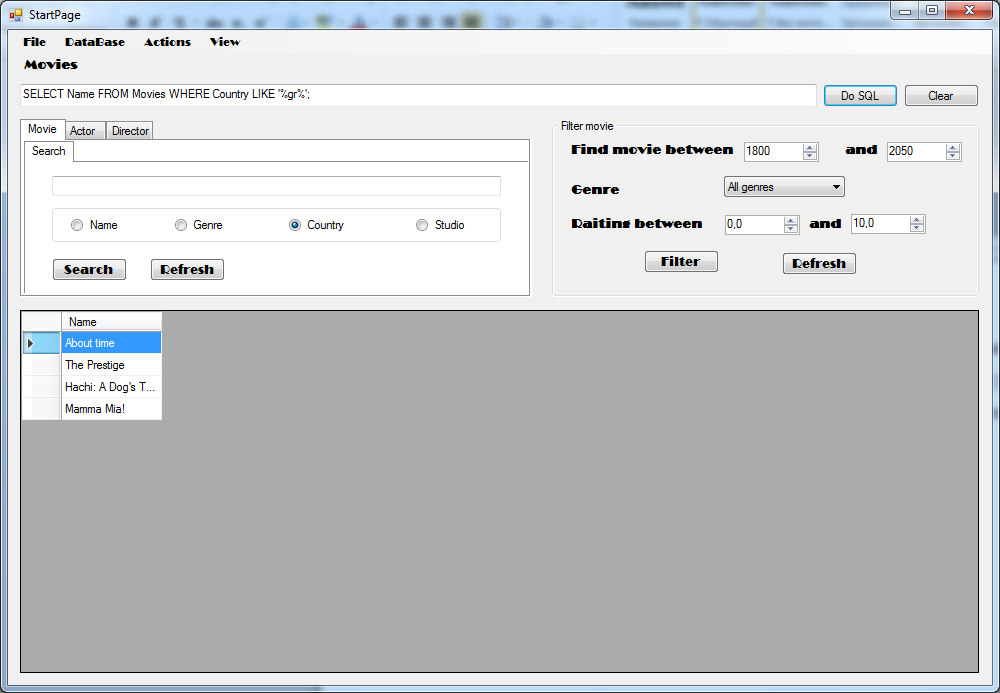


Рисунок 4.13 – Редактор SQL-запросов

Следующая функция программы – формирование отчетов разного формата. При нажатии на кнопку «Save» откроется контекстное меню, в котором нам показаны варианты формата – \*pdf (см. рис. 4.14) или \*doc (см. рис. 4.15).

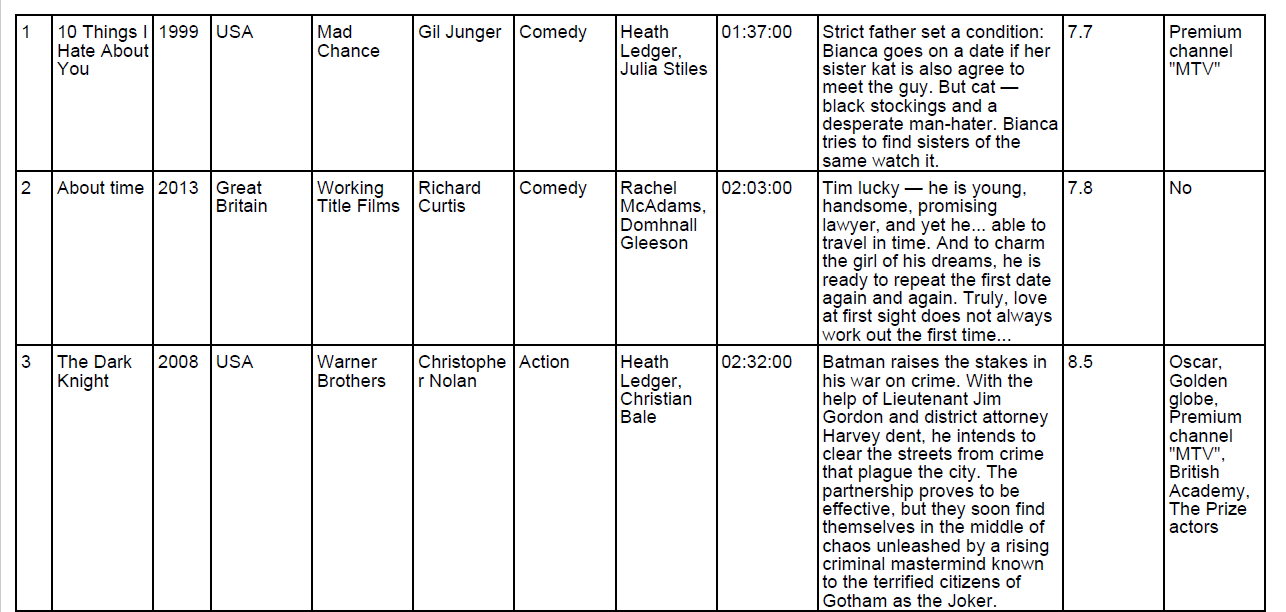


Рисунок 4.14 – Pdf-информация о фильмах

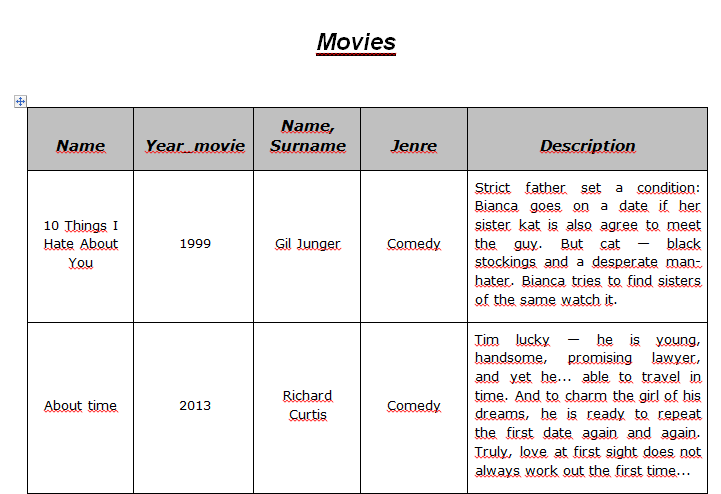


Рисунок 4.15 – Doc-информация о фильмах

Следующая функция – отображение статистической информации. Работу данной функции можно увидеть при нажатии на кнопку меню «View» главного окна, а после на «Statistic», выбрав нужную статистику (см. рис. 4.16).

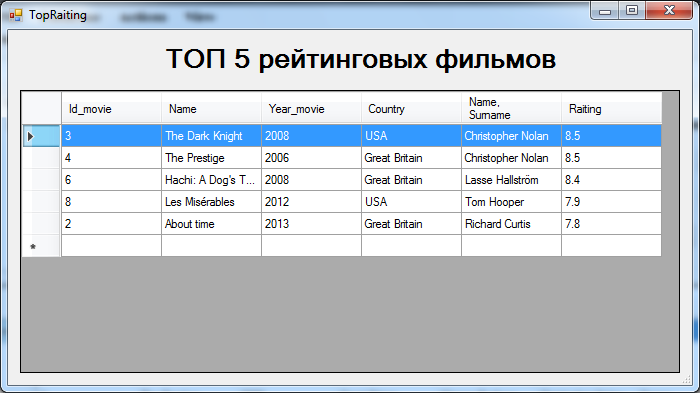


Рисунок 4.16 – Статистика «Топ 5 рейтинговых фильмов»

Запрос статистики «Топ 5 рейтинговых фильмов»:

SELECT TOP 5 [Id\_movie], [Name], [Year\_movie], [Movies].[Country], [Director].[Name, Surname], [Raiting] FROM Movies, Director WHERE Movies.Director = Director.Id\_director ORDER BY Raiting DESC;

Запрос статистики «Количество фильмов по году»:

SELECT TOP 10 Year\_movie, COUNT(Id\_movie) AS 'Movies Count' FROM Movies GROUP BY Year\_movie ORDER BY COUNT(Id\_movie) DESC;

Запрос статистики «Количество фильмов по жанру»:

SELECT TOP 10 Jenre, COUNT(Id\_movie) AS 'Movies Count' FROM Movies GROUP BY Jenre ORDER BY COUNT(Id\_movie) DESC;

4.6 Описание задачи автоматизации

Для того чтобы администратор не тратил слишком много времени на заполнение данных в таблицах, необходимо создать более простой способ их заполнения. Данная задача была решена посредством считывания данных с текстового файла и заполнением таблиц данными. Вид открытия файла представлен на рисунке 4.17.

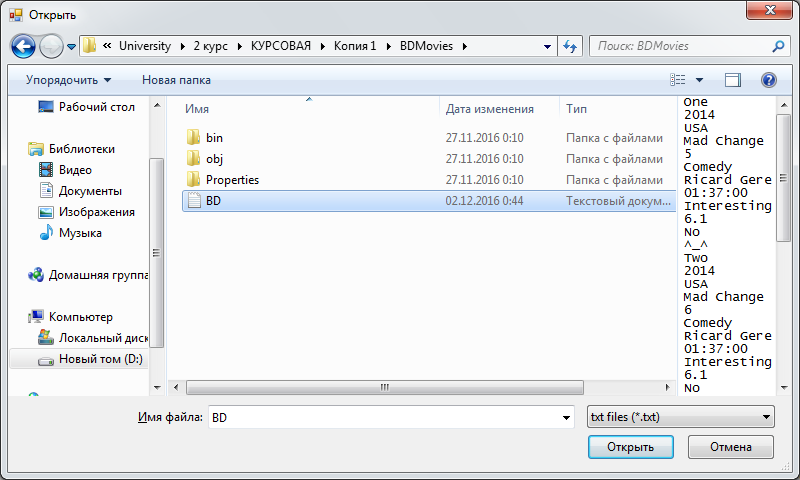


Рисунок 4.17 - Вид открытия файла

Для того чтобы считать информацию, нужно проверить ее на корректность и тогда добавлять в базу. Текстовый файл должен находиться в правильном формате. При открытии файла, в списке фильмов будет отображена новая информация, не смотря на данные, хранящиеся в БД. Результат работы функции можно увидеть на рисунке 4.18.

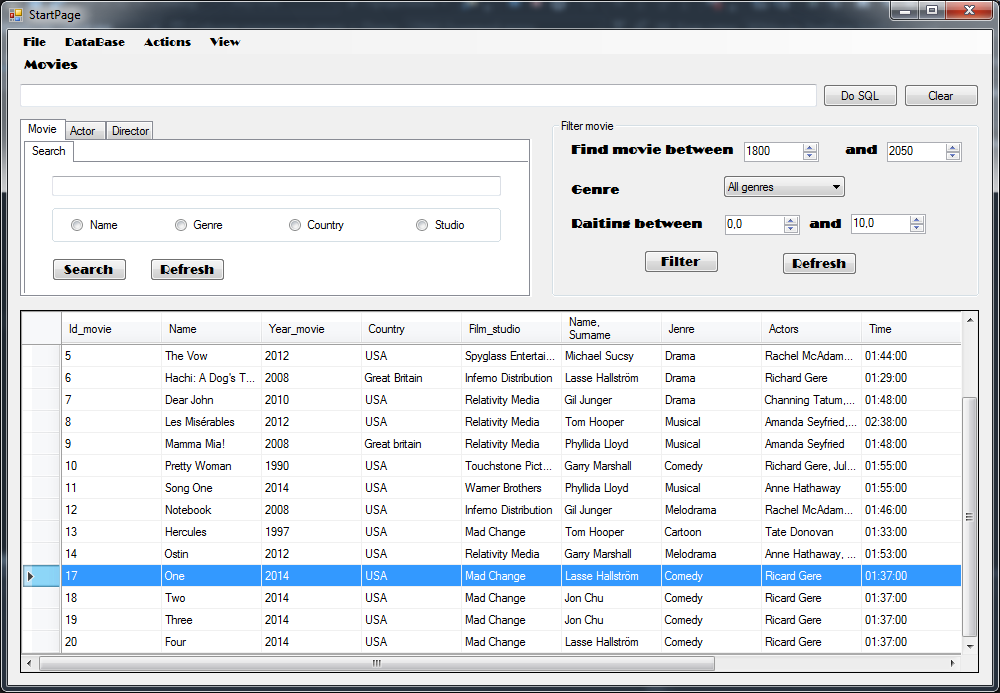


Рисунок 4.18 - Результат работы функции

Ниже представлен код, с помощью которого происходит процесс автоматизации:

private void openToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

Stream myStream = null;

openFileDialog1.InitialDirectory = "d:\\university\\2 курс\\КУРСОВАЯ";

openFileDialog1.Filter = "All files (\*.\*)|\*.\*|txt files (\*.txt)|\*.txt";

openFileDialog1.FilterIndex = 2;

openFileDialog1.RestoreDirectory = true

if (openFileDialog1.ShowDialog() == DialogResult.OK)

{

try

{

if ((myStream = openFileDialog1.OpenFile()) != null)

{

using (myStream)

{

string[] lines = File.ReadAllLines(openFileDialog1.FileName);

for (int i = 0; i < lines.Length; i += 12)

{

string name = lines[i];

string year = lines[i + 1];

string country = lines[i + 2];

string studio = lines[i + 3];

string director = lines[i + 4];

string genre = lines[i + 5];

string actors = lines[i + 6];

string time = lines[i + 7];

string desc = lines[i + 8];

string rate = lines[i + 9];

string awards = lines[i + 10];

string query = String.Format(@"INSERT INTO Movies VALUES('{0}', '{1}', '{2}', '{3}',

'{4}', '{5}', '{6}', '{7}', '{8}', '{9}', '{10}')",

name, year, country, studio, director, genre, actors, time, desc, rate, awards);

SqlDataAdapter oda = new SqlDataAdapter(query, con);

DataTable dt = new DataTable();

oda.Fill(dt);

con.Close();

}

MessageBox.Show((lines.Length + 1) / 12 + " movies were added successfully");

}

}

StartPage\_Load(null, null);

}

catch (Exception ex)

{

MessageBox.Show("Error: Could not read file from disk. Original error: " + ex.Message);

}

}

}

ВЫВОДЫ

В результате выполнения курсовой работы была разработана информационная система «Кинофильмы». Курсовая работа включает в себя создание базы данных и клиентской программы, которая взаимодействует с базой данных. Были изучены основы работы с базами данных, получен опыт разработки и проектирования баз данных, а также разработки программных продуктов с использованием баз данных. Были закреплены знания в области sql, нормализации баз данных и моделирования.

В ходе работы также были проанализированы предметная область, проблемы и задачи, которые существуют в данной области, а также то, каким образом наличие данной информационной системы позволит их решить или облегчить их выполнения для сотрудников.

Для реализации программы была использована платформа Visual Studio 2015, технология Windows Forms и язык программирования С#. В качестве системы управления базами данных была выбрана MSSQL.

Были реализованы все функции, которые были указаны при постановке задачи. Данный программный продукт позволяет выполнить такие операции, как поиск необходимой информации, фильтрация данных, добавление, редактирование, удаление данных в таблицах. Также программа позволяет совершать более сложные операции, такие как формирование отчета, получение необходимой статистики и добавление новых фильмов с помощью текстового документа.

По моему мнению, разработанная ИС может значительно упростить жизнь студентам, которые изучают киноискусство или просто им увлекаются. В ней присутствует реализация как самых простых и необходимых функций, так и более сложных, что сделает работу предприятия более гибкой из-за уменьшения проблем при работе с документооборотом.

ПЕРЕЧЕНЬ ССЫЛОК

1. К. Дж. Дейт Введение в системы баз данных. 8-е изд. [Текст] /Дейт К. Дж.– М.:Издательскийдом„Вильямс”, 2005.- 1328с.

2. Мартин Грабер. SQL [Текст]/ Мартин Грабер. - К.: Изд-во “ЛОРИ”, 2003. - 644 с..

3. Гарсия-Молина Системы баз данных. Полный курс [Текст] / Гарсия-Молина, Гектор, Ульман, Джеффри, Уидом, Дженнифер.: Пер. с англ. – М.:Издательскийдом„Вильямс”, 2003. – 1088 с.

4. Грофф Дж. Р.SQL. Полное руководство [Текст] / Грофф Дж. Р., Пол Н. Вайнберг.: пер. с англ. – СПб.: BHV, 2001. – 816 с.

5. Д.Кренке. Теория и практика построения баз данных. 8-е изд. [Текст]/ Д.Кренке. – СПб.: Питер, 2003. – 800 с.: ил.