**Введение**

На сегодняшний день базы данных, содержащие списки фильмов, музыки, книг и прочих развлечений набирают всё большую и большую популярность.

Пользователи интернета используют подобные сервисы для того, чтоб найти фильм для просмотра вечером, узнать отзывы других людей о вышедшей на экраны новинке, поделиться своим мнением о прочитанной накануне книге, вести свой список просмотренных фильмов и мультфильмов, поставить оценку и посетовать на то, что у популярной группы новый альбом вышел слабее, чем предыдущие.

С развитием интернета, существование подобных сервисов серьезно повлияло и на авторов. Теперь количество проданных книг зависит не только от размера средств, потраченных на рекламу, но и от того, насколько тепло произведение было воспринято читателями. Если раньше среднестатистический зритель, выбирая, на какой фильм пойдёт в кинотеатр, мог опираться только на советы знакомых, кассира в кинотеатре или на яркий постер, висящий около его дома, то на сегодняшний день, используя глобальную сеть, он может узнать мнение сотен тысяч человек по всему миру. Данная тенденция заставляет авторов куда серьезнее подходить к созданию своих произведений, продумывать каждую деталь, внимательнее относиться к мелочам, так как каждый прокол обязательно будет замечен внимательным зрителем, а это, в свою очередь, обязательно повлияет на продажи произведения.

В англоязычном сегменте интернета самыми популярными сайтами подобной тематики являются следующие сервисы:

* «IMDb» — сервис, позволяющий узнать обо всех фильмах мира;
* «Metacritic» — сайт, позволяющий узнать отзывы о музыкальных альбомах, играх, фильмах, телевизионных шоу и DVD-дисках;
* «Rotten Tomatoes» — веб-сайт, на котором собираются обзоры информация и новости кинематографа, позволяющий «кинуть помидором» в непонравившийся фильм.

В русском Интернете наиболее популярным сайтом в данном сегменте является «Кинопоиск» — русскоязычный аналог IMDb, позволяющий оценивать фильмы. На сегодняшний день не существует ни одного популярного сайта, позволяющего оценивать книги и музыку.

Целью данной курсовой работы является создание веб-приложения, позволяющего пользователям находить информацию, оценивать, писать рецензии к фильмам, музыке и книгам. Система также должна предоставлять возможность пользователям делиться информацией об оцененных произведениях.

Согласно цели, были поставлены следующие задачи:

* Анализ предметной области;
* Разработка схемы базы данных;
* Реализация разработанной схемы в конкретной СУБД (MySQL);
* Создание удобного для использования интерфейса веб-сайта;
* Написание кода реализующего взаимосвязь интерфейса и базы данных.

Для разработки были выбраны следующие языки технологии:

* HTML — язык гипертекстовой разметки. Необходим для создания структуры интернет-сайта системы;
* CSS — каскадные таблицы стилей. Необходимы для задания внешнего вида интернет-сайту системы;
* PHP — язык программирования, применяемый для разработки веб-приложений;
* SQL — язык, применяемый для создания, модификации и управления данными в реляционной базе данных.
* Реляционная система управления базами данных MySQL.

**Анализ предметной области**

1. *Описание предметной области*

Каждая раздел системы, помимо уникальной части, должен иметь следующую общую часть:

* Главное меню, позволяющее пользователю перейти на другие доступные ему страницы сайта:

- Раздел «Фильмы», позволяющий искать, добавлять и редактировать свои списки просмотренных фильмов, а также информацию о них;

- Раздел «Книги», позволяющий искать, добавлять и редактировать свои списки прочитанных книг, а также информацию о них;

- Раздел «Музыка», позволяющий искать, добавлять и редактировать свои списки музыки, а также информацию о них;

- Раздел «Сообщество», позволяющий просматривать всех пользователей, а также имеющий ссылку на страницу текущего пользователя

* Строка приветствия с именем текущего пользователя. Имя пользователя должно являться ссылкой на его страницу;
* Подвал сайта, выводящий информацию о создателе.

Страницы системы:

1. Главная страница — должна выводить краткую информацию о трех последних фильмах, книгах и песнях, добавленных на сайт вместе со ссылками на страницу с полной информацией.
2. Все фильмы/книги/музыка — страница, предназначенная для просмотра всех произведений определённого типа. Максимальное кол-во произведений на странице не должно превышать 10. Если на сайте добавлено более 10 произведений, то внизу должен появиться переключатель страниц.
3. Страница пользователя — должна выводить информацию о пользователе, а также список его добавленных и оцененных книг, фильмов, музыки.
4. Страница конкретного произведения — должна выводить полную информацию о выбранном произведении, а также оценки и комментарии других пользователей. Должна быть реализована возможность оставить свою оценку и добавить комментарий.
5. Добавить фильм/книгу/музыку — форма добавления нового произведения. Должна содержать ссылку на открытие формы добавления автора произведения. Должна содержать поля для ввода названия, года, автора, жанра, краткого описания и постера произведения.
6. Добавить автора — форма, необходимая для добавления режиссёра/писателя/композитора. Должна содержать поля для ввода Имени и фамилии автора, а также его фотографии.
7. *Сбор данных об объектах предметной области*

Основываясь на информации (п. 1.2) были выявлены основные объекты базы данных, их атрибуты, форматы ограничения на значения, первичные ключи. Результаты приведены в таблицах:

## Структура таблицы user

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***user\_id*** | int(11) | Нет |  |
| login | varchar(20) | Нет |  |
| password | varchar(32) | Нет |  |
| name | varchar(30) | Да | NULL |
| surname | varchar(30) | Да | NULL |
| email | varchar(50) | Нет |  |
| register\_time | datetime | Нет | CURRENT\_TIMESTAMP |
| salt | int(3) | Нет |  |

## Структура таблицы movies

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***movie\_id*** | int(11) | Нет |  |
| title | varchar(100) | Нет |  |
| type | char(1) | Нет |  |
| poster | varchar(100) | Да | NULL |
| director | int(11) | Да | NULL |
| year | year(4) | Да | NULL |
| genre | varchar(70) | Да | NULL |
| rating | int(11) | Да | NULL |
| rated\_by | int(4) | Да | NULL |
| description | blob | Нет |  |

## Структура таблицы movie\_rating

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| movie | int(11) | Нет |  |
| user | int(11) | Нет |  |
| comment | blob | Да | NULL |
| rating | tinyint(2) | Да | NULL |

## Структура таблицы directors

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***director\_id*** | int(11) | Нет |  |
| name | varchar(30) | Нет |  |
| surname | varchar(30) | Нет |  |
| photo | varchar(70) | Да | NULL |

## 

## Структура таблицы books

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***book\_id*** | int(11) | Нет |  |
| title | varchar(100) | Нет |  |
| type | char(1) | Нет |  |
| logo | varchar(100) | Да | NULL |
| writer | int(11) | Да | NULL |
| year | year(4) | Да | NULL |
| genre | varchar(70) | Да | NULL |
| rating | int(11) | Да | NULL |
| rated\_by | int(4) | Да | NULL |
| description | blob | Нет |  |

## Структура таблицы book\_rating

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| book | int(11) | Нет |  |
| user | int(11) | Нет |  |
| comment | blob | Да | NULL |
| rating | tinyint(2) | Да | NULL |

## Структура таблицы writers

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***writer\_id*** | int(11) | Нет |  |
| name | varchar(30) | Нет |  |
| surname | varchar(30) | Нет |  |
| photo | varchar(70) | Да | NULL |

## Структура таблицы music

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***music\_id*** | int(11) | Нет |  |
| title | varchar(100) | Нет |  |
| type | char(1) | Нет |  |
| logo | varchar(100) | Да | NULL |
| musician | int(11) | Да | NULL |
| year | year(4) | Да | NULL |
| genre | varchar(70) | Да | NULL |
| rating | int(11) | Да | NULL |
| rated\_by | int(4) | Да | NULL |
| description | blob | Нет |  |

## Структура таблицы music\_rating

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| music | int(11) | Нет |  |
| user | int(11) | Нет |  |
| comment | blob | Да | NULL |
| rating | tinyint(2) | Да | NULL |

## Структура таблицы musicians

|  |  |  |  |
| --- | --- | --- | --- |
| **Столбец** | **Тип** | **Null** | **По умолчанию** |
| ***musician\_id*** | int(11) | Нет |  |
| **type** | char(1) | Нет |  |
| name | varchar(30) | Нет |  |
| surname | varchar(30) | Да |  |
| photo | varchar(70) | Да | NULL |

1. *Выявление и определение типа связей между объектами и построение информационно-логической модели.*

В результате анализа были выявлены и определены следующие связи между элементами таблицы:



**Разработка логической структуры БД**

1. *Представление и анализ структуры базы данных.*

База данных (БД) – это совокупность взаимосвязанных, хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений.

Создание базы данных, ее поддержка и обеспечение доступа пользователей к ней осуществляется централизованно с помощью специального программного инструментария – системы управления базами данных.

Система управления базами данных (СУБД) – это комплекс программных и языковых средств, необходимых для создания баз данных, поддержания их в актуальном состоянии и организации поиска в них необходимой информации.

Наиболее распространёнными моделями баз данных являются иерархическая, сетевая и реляционная.

**Иерархическая структура** представляет совокупность элементов, связанных между собой по определенным правилам. Объекты, связанные иерархическими отношениями, образуют ориентированный граф (перевернутое дерево), вид которого представлен на рисунке:

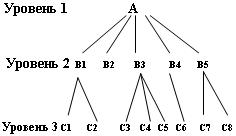


Рисунок 3.1. Иерархическая модель данных

Основные понятия иерархической структуры − это узел, уровень и связь.

Узел – это совокупность атрибутов данных, описывающих некоторый объект. На схеме иерархического дерева узлы представляются вершинами графа. Каждый узел на более низком уровне связан только с одним узлом, находящимся на более высоком уровне.

Иерархическое дерево имеет только одну вершину (корень дерева), не подчиненную никакой другой вершине и находящуюся на самом верхнем (первом) уровне. Зависимые (подчиненные) узлы находятся на втором, третьем и т.д. уровнях. К каждой записи базы данных существует только один (иерархический) путь от корневой записи. Например, как видно на рисунке (рис. 3.1), для записи С4 путь проходит через записи ВЗ к А.

В **сетевой** **структуре** при тех же основных понятиях (уровень, узел, связь) каждый элемент может быть связан с любым другим элементом (рис. 3.2).

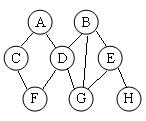


Рисунок 3.2. Иерархическая модель данных

