

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по научно-исследовательской работе
Тема: «Разработка WEB-приложений и ASP.NET»

Студент гр. 5304

Маслов Н.Д.

Руководитель

Попова Е.В.

Санкт-Петербург

2020

**ЗАДАНИЕ
НА НАУЧНО-ИССЛЕДОВАТЕЛЬСКУЮ РАБОТУ**

Студент Маслов Н.Д.

Группа 5304

Тема НИР: Разработка WEB-приложений и ASP.NET

Задание на НИР:

Разработка WEB-приложения и WEB-сервиса используя платформу ASP.NET. Провести сравнительный анализ.

Сроки выполнения НИР: 01.12.2019 – 01.06.2021

Дата сдачи отчета: 01.06.2020

Дата защиты отчета: 15.06.2021

Студент

Маслов Н.Д.

Руководитель

Попова Е.В.

АННОТАЦИЯ

В данной научно-исследовательской работе будет продемонстрирован процесс разработки WEB-приложения и WEB-сервиса с использованием платформы ASP.NET. Так же будет проведён сравнительный анализ особенностей разработки двух разных программных продуктов.

SUMMARY

This research work will demonstrate the process of developing a WEB-application and WEB-service using the ASP.NET platform. There will also be a comparative analysis of the development features of two different software products.

Содержание

АННОТАЦИЯ	3
Введение	5
ПОСТАНОВКА ЗАДАЧИ	6
1.1. Актуальность.....	6
1.2. Проблема	7
1.3. Цель.....	7
1.4. Задачи	7
РЕЗУЛЬТАТЫ РАБОТЫ В ОСЕННЕМ СЕМЕСТРЕ	8
1 Разработка WEB-приложения и WEB-сервиса.....	8
ПЛАНЫ РАБОТЫ НА ВЕСЕННИЙ СЕМЕСТР	14
Список использованных источников.....	15
Приложение А	16

Введение

ASP.NET в настоящее время является одной из самых популярных и перспективных платформ. Это кроссплатформенная высокопроизводительная среда с открытым исходным кодом. При разработке с помощью ASP.NET используется паттерн MVC, который соответствует архитектурному шаблону: модель – представление – контроллер.

Для взаимодействия клиента с сервером используется протокол HTTP – проток передачи гипертекста. В своей основе этот протокол является текстовым, что делает его простым в понимании, удобным в поддержке, отладке, а также удобным для межплатформенного взаимодействия.

Целью данной работы является исследование методов взаимодействия между клиентом и сервером. Для проведения исследований необходимо разработать WEB-приложение и WEB-сервис с использованием среды ASP.NET.

ПОСТАНОВКА ЗАДАЧИ

1.1. Актуальность

В мире для разработки веб-приложений и веб сервисов существует огромное множество различных фреймворков и языков. У каждой платформы есть свои особенности и преимущества.

ASP.NET – это серверная платформа веб-приложений с открытым исходным кодом, разработанная Microsoft для Windows и запущенная в начале 2000-х годов.

ASP.NET обладает множеством преимуществ, такими как высокая скорость, низкая стоимость, обширная поддержка. Веб-сайты и приложения, созданные с помощью ASP.NET, могут быть быстрее и эффективнее, чем, например, создание веб-сайтов с помощью PHP. Приложения ASP.NET компилируются, что означает, что код переводится в объектный код, который затем выполняется. Этот процесс компиляции занимает мало времени, но происходит только один раз. После компиляции код может очень быстро выполняться платформой .Net снова.

Интерпретируемый код не выполняется непосредственно машиной, но его необходимо читать и интерпретировать каждый раз перед выполнением. Скомпилированный код обычно быстрее и масштабируемое, чем интерпретируемый код, и может делать все, что может делать интерпретируемый код. Примеры интерпретируемых языков включают PHP, JavaScript и Ruby.

Процесс компиляции также обеспечивает проверку согласованности всего кода. Например, если метод с именем GetUser переименован в GetEmployee как часть некоторых обновлений кода, любая ссылка на GetUser в остальной части приложения вызовет ошибку во время компиляции, что упрощает идентификацию и исправление. Интерпретируемые языки не выявят эту ошибку, пока код не будет запущен и протестирован. В большом приложении очень много времени требуется для ручного тестирования каждого сценария

или для написания и поддержки дополнительного кода, который проверяет каждый сценарий при каждом изменении кода.

1.2. Проблема

При разработке больших приложений остро стоит вопрос корректного взаимодействия клиентской и серверной части приложений. Для того что бы приложение быстро и без ошибок выполняло возложенные на него функции необходимо грамотно использовать методы взаимодействия двух частей приложения.

1.3. Цель

В рамках работы необходимо создать WEB-приложение и WEB-сервис и исследовать методы их взаимодействия

1.4. Задачи

Для выполнения поставленных целей в научно-исследовательской работе необходимо разработать WEB-приложение. В качестве приложения был выбран сервис отображающие функции аренды фильмов.

РЕЗУЛЬТАТЫ РАБОТЫ В ОСЕННЕМ СЕМЕСТРЕ

1 Разработка WEB-приложения и WEB-сервиса

Были реализованы модели фильмов, юзеров, ролей и жанров, ниже представлены их диаграммы.

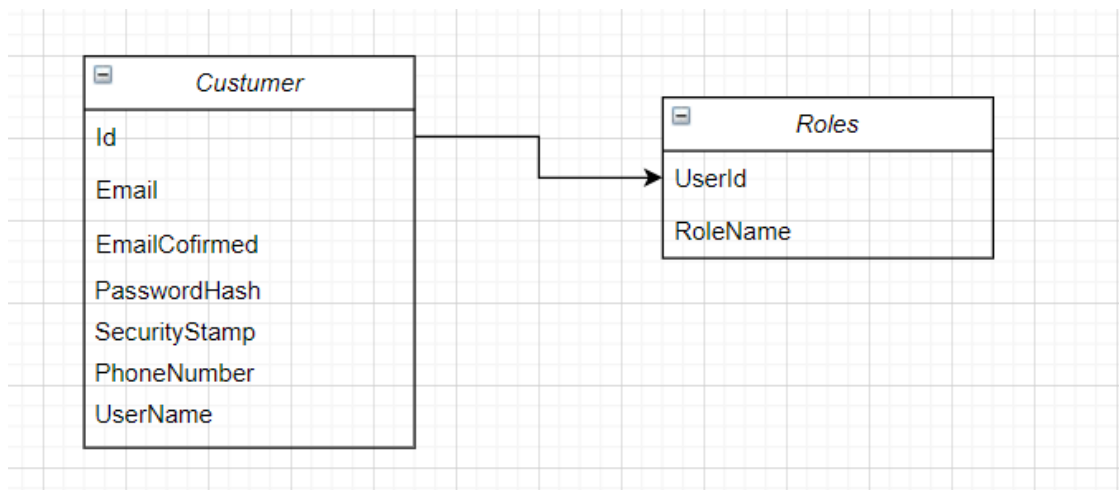


Рисунок 1 - Модели пользователя и ролей

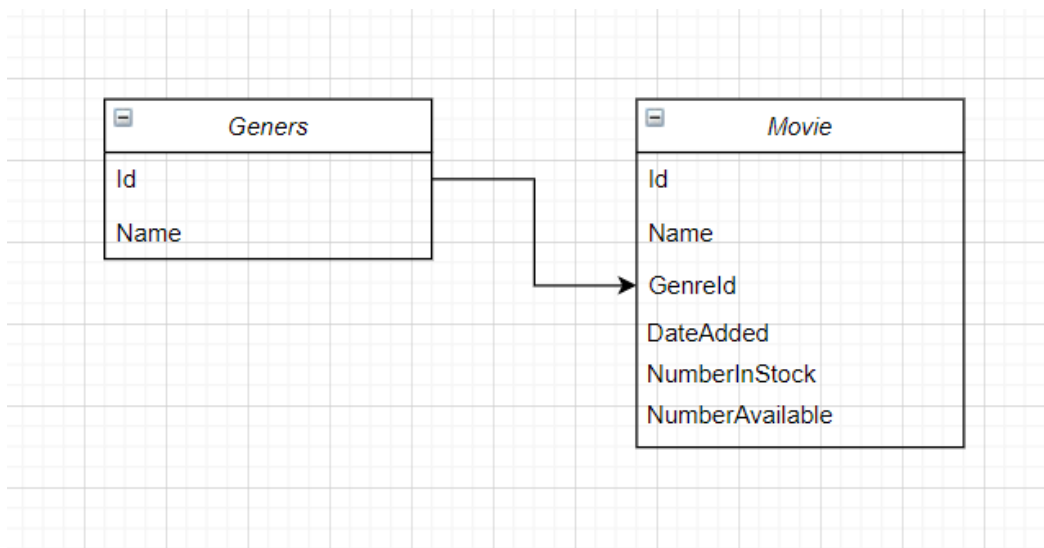


Рисунок 2 - Модели фильма и жанров

С главной страницы был убран не относящийся к приложению контент, её конечный вид представлен на рисунке 3.

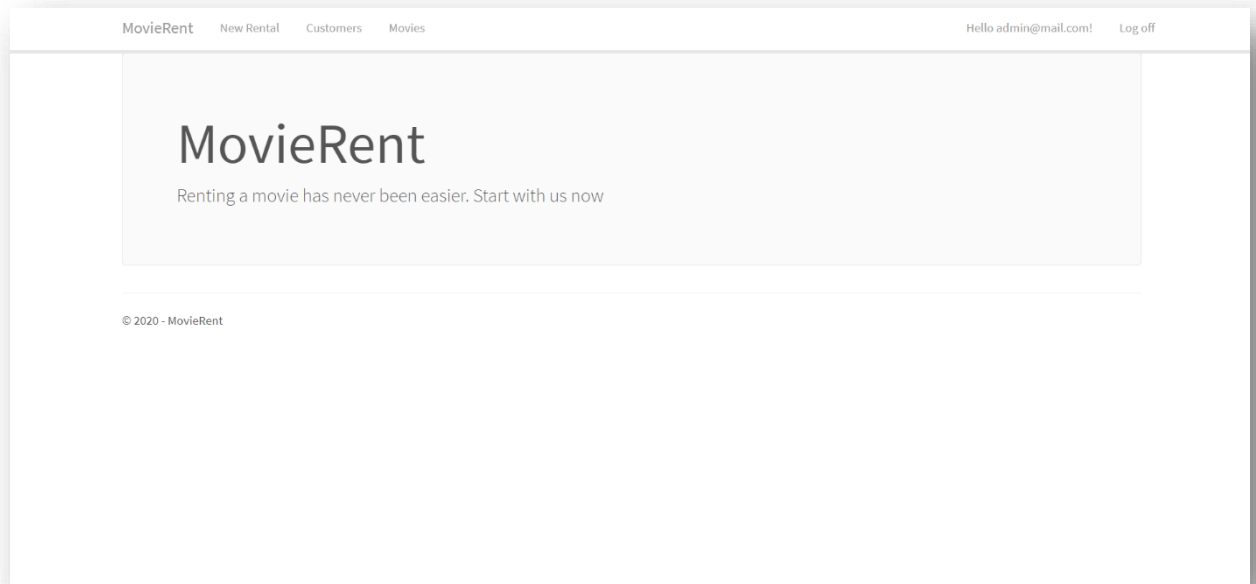


Рисунок 3 - Главная страница

Так же были доделаны страницы существующих пользователей и доступных фильмов рисунок 4 и 5.

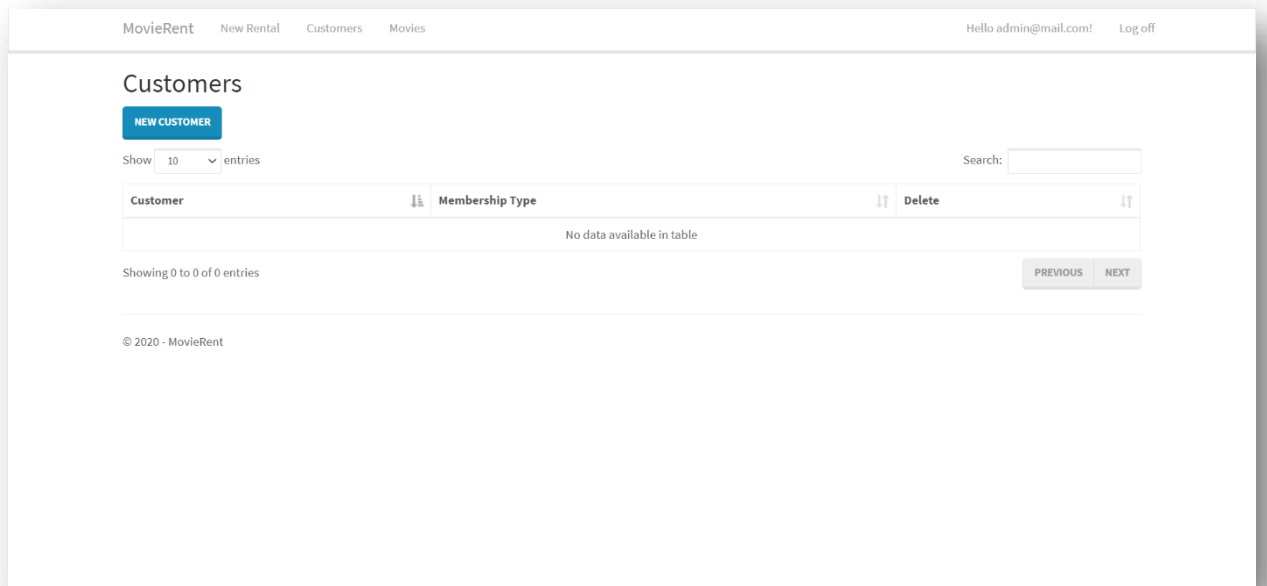
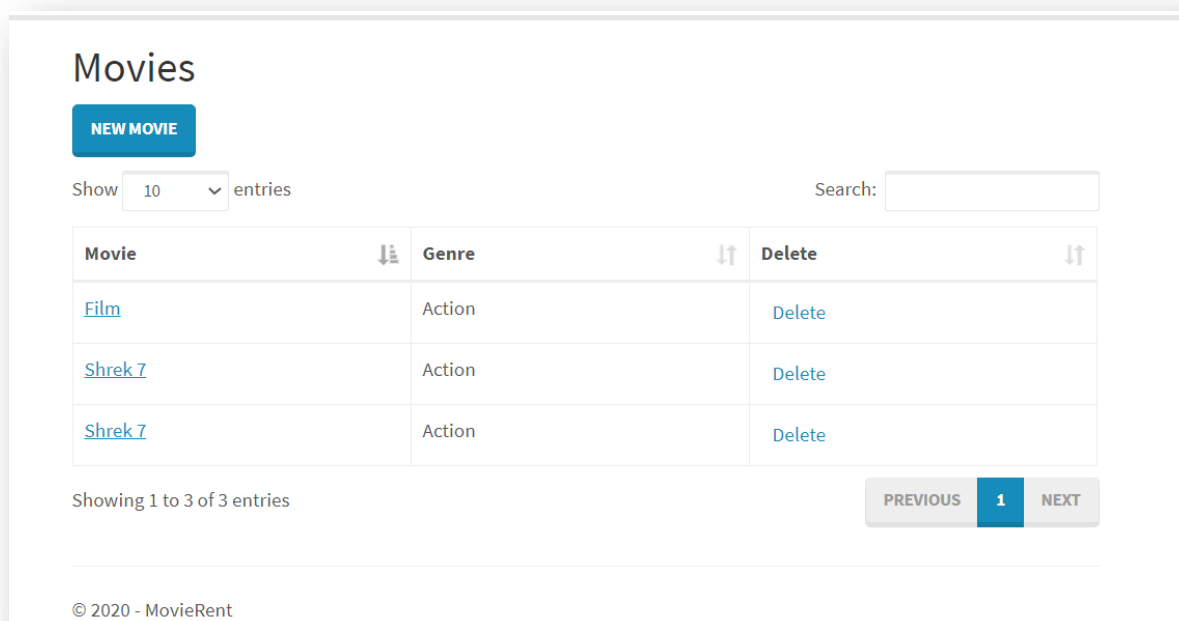


Рисунок 4 - Страница пользователей



Рисунки 6, 7, 8 – Процесс добавления фильма

Так же были добавлены возможности редактирования фильмов и их удаление. Рисунки 9 и 10.

Edit Movie

Name

Film

Release Date

24 ноя 2020

The field Release Date must be a date.

Genre

Thriller

Number in Stock

2

SAVE

© 2020 - MovieRent

Рисунок 9 – Редактирование фильма

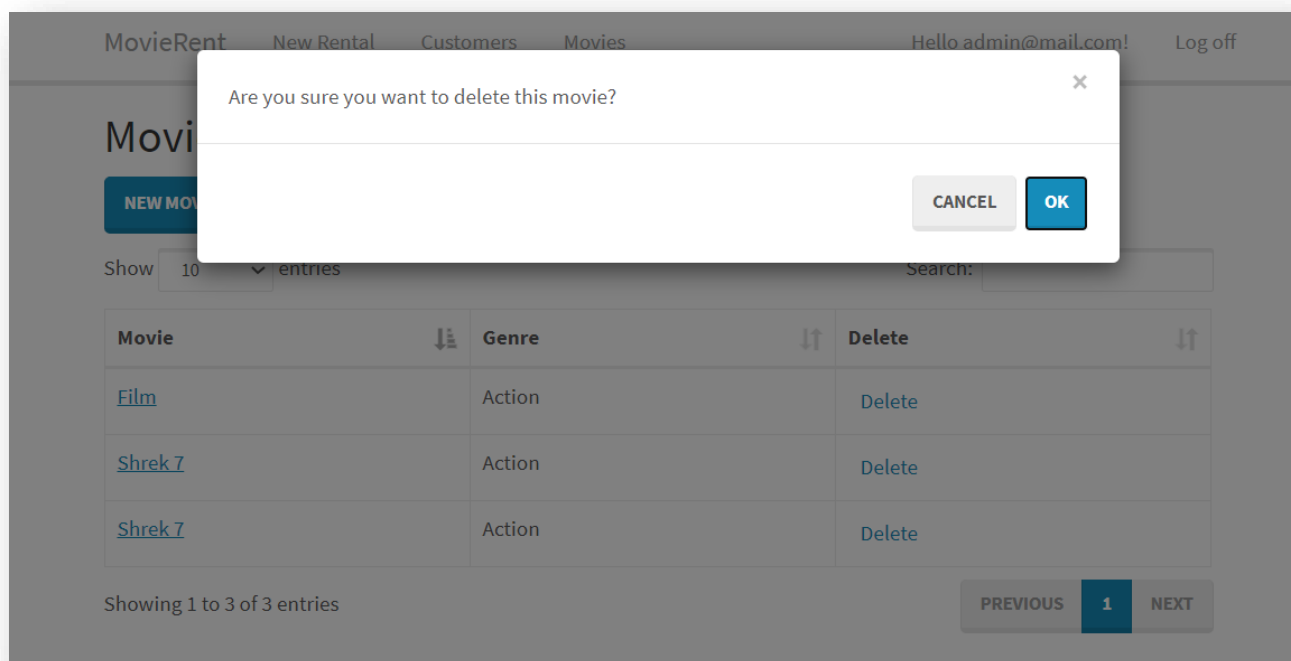


Рисунок – 10 Удаление фильма

Аналогично было сделано для добавления редактирования и удаления пользователей. Помимо этого были сделаны права на такие действия как создание, удаление, редактирование фильмов пользователей и добавлены проверки на роль обладающую нужными правами.

Для функционирования приложения был написан REST api со следующими роутингами.

Пути пользователей:

- GET /api/customers
- GET /api/customer/:id
- POST /api/customer/:id
- DELETE /api/customer/:id

Пути фильмов:

- GET /api/movies
- GET /api/movie/:id
- POST /api/ movie/:id
- DELETE /api/ movie/:id

ПЛАНЫ РАБОТЫ НА ВЕСЕННИЙ СЕМЕСТР

1. Пофиксить баги блокирующие работу приложения.
2. Доделать API, возможность становления пользователя клиентом и выбирать тарифы.
3. Провести исследования методов API и сравнительный анализ разработки сервиса и приложения на ASP.NET.

Список использованных источников

1. Троелсон Э. С# и платформа .Net / пер. с английского ЗАО Издательский дом «Питер» 2004г. 782 с.: ил.
2. Троелсон Э., Джепкинск Ф. Язык программирования С# и платформы .Net и .Net Core / пер. с английского ООО «Диалектика» 2018г. 1316 с.: ил.
3. Электронный ресурс <https://habr.com/ru/post/175999/>

Приложение А

Ссылка на GitHub репозиторий с кодом проекта

<https://github.com/NikitaMasl/MovieRent>