Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

WEB-ПРИЛОЖЕНИЕ «САЙТ КИНОТЕАТРА»

отчет по дисциплине

«Научно-исследовательская работа студентов 2»

Выполнил:

студент гр. 589-1

\_\_\_\_\_\_\_\_\_ Жданова А.А.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_2022 г.

Проверил:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_Черкашин М.В.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_2022 г.

Томск 2022

**Содержание**

[ВВЕДЕНИЕ 3](#_Toc123047830)

[1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 5](#_Toc123047831)

[1.1 Принципы работы веб-приложений 5](#_Toc123047832)

[1.2 Доступ к базе данных 6](#_Toc123047833)

[1.3 Преимущества веб-приложений 8](#_Toc123047834)

[2. ПРОБЛЕМАТИКА 9](#_Toc123047835)

[3. ПОСТАНОВКА ЗАДАЧИ 12](#_Toc123047836)

[4. ВЫБОР СРЕДСТВ РЕАЛИЗАЦИИ 14](#_Toc123047837)

[5. ПРОЕКТ ПРЕДЛАГАЕМОГО РЕШЕНИЯ 16](#_Toc123047838)

[5.1 Реализация 16](#_Toc123047839)

[5.2 Результаты программы 19](#_Toc123047840)

[Заключение 29](#_Toc123047841)

[Список используемых источников 30](#_Toc123047842)

# **ВВЕДЕНИЕ**

Автоматизация – это уменьшение (или полное исключение) участия человека в различных процессах с помощью математических и технических средств. Зачастую это повышает производительность, улучшает качество результата и оптимизирует сам процесс, позволяя более рационально использовать необходимые ресурсы [1].

Постоянно развивающийся технический прогресс толкает производство к постоянному и непрерывному изменению, путем обновления способов изготовления продукта, появлением новых автоматизированных линий, позволяющих процессу производства становиться более емким и технологичным. Современные машины и оборудование позволяют до минимума сократить сроки изготовления продукции, а также улучшить ее качество.

В условиях автоматизированного производства наиболее актуальной задачей является комплекс мероприятий по разработке технологических процессов и созданию на их основе высокопроизводительных машин. Сегодня, автоматизация производства представляет собой одно из перспективных направлений в развитии промышленности.

Однако, несмотря на очевидную прогрессивность автоматизации и повышенное внимание к ней, её внедрение осуществляется замедленными темпами на современных предприятиях [2].

Данная работа направлена на разработку веб-приложения для сети антикинотеатров. Веб-приложения на данный момент являются и еще долгое время будут являться актуальными среди абсолютного большинства пользователей по причине быстроты и удобства использования.

Зайти на сайт и сделать заказ проще, чем ехать в магазин, стоять в очереди и нести домой тяжелую покупку. С помощью функции поиска пользователь может найти необходимый товар за считанные секунды и не тратить время на просмотр всех позиций вручную. Веб-приложения превратили шопинг из многочасового и выматывающего занятия в покупку в два клика, при этом товар доставят покупателю прямо к входной двери.

По сути, веб-приложение — это программа, однако, она не требует установки на смартфон или компьютер, а доступ к ней пользователь получает через браузер. Информация в веб-приложении может обновляться удаленно автоматически, при этом постоянное обновления HTML-страниц сайта не требуется.

Так же стоит отметить, что веб-приложения позволяют собирать, сохранять и анализировать данные, полученные от посетителей сайта, то есть владельцы сайта всегда будут в курсе предпочтений своей аудитории, что позволит менять содержимое самого сайта и улучшать бизнес, тем самым увеличивая прибыль компании.

**Цель работы:** создание веб-приложения для сети антикинотеатров.

Выделенными **задачами** являются:

* Автоматизирование процесса выбора и покупки билетов для клиентов;
* Создание интерфейса сайта;
* Разделение фильмов по жанрам;
* Создание базы данных для хранения данных пользователей;
* Разработка дизайна сайта;
* Проектирование формы для оформления заказа;
* Добавление регистрации пользователей.

# **ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ**

Веб-приложение — это приложение, одна часть которого загружается в браузер и взаимодействует с пользователем (визуально-интерфейсная часть), а другая находится на веб-сервере и выполняет запросы, поступающие от первой, а затем возвращает ответ. Часть, которая загружается в браузер и с которой взаимодействует пользователь, называется клиентской частью (фронтенд). На веб-сервере находится серверная часть веб-приложения (бэкенд) [3].

# **1.1 Принципы работы веб-приложений**

Любое веб-приложение представляет собой набор статических и динамических веб-страниц. *Статическая веб-страница* — это страница, которая всегда отображается перед пользователем в неизменном виде. Веб-сервер отправляет страницу по запросу веб-браузера без каких-либо изменений. В противоположность этому, сервер вносит изменения в динамическую веб-страницу перед отправкой ее браузеру. По причине того, что страница меняется, она называется динамической.

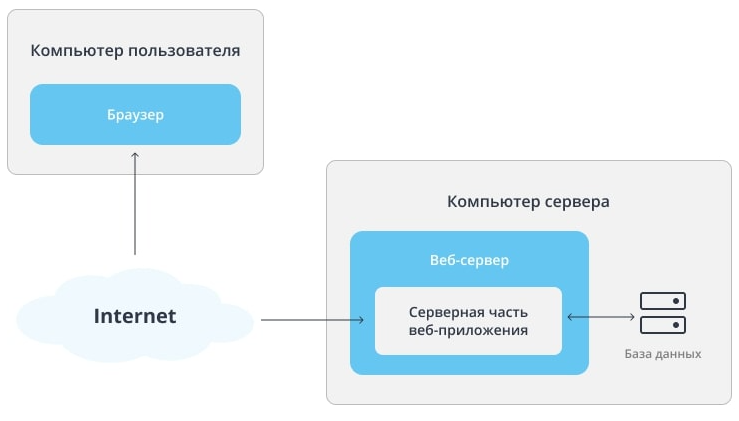


Рисунок 1.1 - Принципы работы веб-приложений

Например, можно создать страницу, на которой будут отображены результаты программы оздоровления. При этом некоторая информация (например, имя сотрудника и его результаты) будет определяться в момент запроса страницы сотрудником [4].

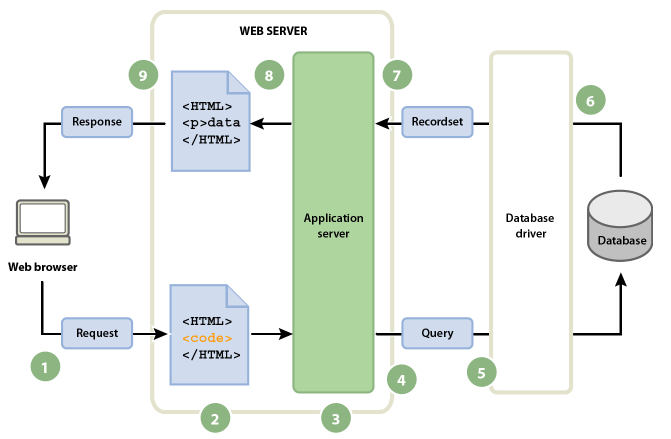
# **1.2 Доступ к базе данных**

Сервер приложений предоставляет возможность использовать такие ресурсы сервера, как базы данных. Например, динамическая страница может содержать программные инструкции для сервера приложений, следуя которым серверу необходимо получить определенные данные из базы данных и поместить их в HTML-код страницы.

Хранение содержимого в базе данных позволяет отделить оформление веб-сайта от содержимого, которое будут видеть пользователи. Вместо того чтобы создавать все страницы в виде отдельных HTML-файлов, пишутся только шаблоны страниц для каждого вида представляемой информации. Затем содержимое загружается в базу данных, после чего веб-сайт будет извлекать его при запросах пользователей. Кроме того, можно обновить информацию в одном источнике и продублировать это изменение на всем веб-сайте без редактирования каждой страницы вручную.

Программная инструкция, предназначенная для получения данных из базы данных, называется запросом к базе данных. Запрос состоит из критериев поиска, выраженных с помощью языка баз данных, называемого SQL (язык структурированных запросов). Текст SQL-запроса располагается в сценариях страниц на стороне сервера либо в тегах.

После того как драйвер установит соединение, выполняется запрос к базе, в результате чего формируется набор записей. Набор записей представляет собой множество данных, полученных из одной или нескольких таблиц базы данных. Набор записей возвращается серверу приложений, который использует полученные данные для формирования страницы.

Рисунок 1.2 - Доступ к базе данных

**1.** Веб-браузер запрашивает динамическую страницу. **2.** Веб-сервер находит страницу и передает ее серверу приложений. **3.** Сервер приложений просматривает страницу на наличие инструкций и выполняет ее подготовку. **4.** Сервер приложений отправляет запрос драйверу базы данных. **5.** Драйвер выполняет запрос в базе данных. **6.** Драйверу возвращается набор записей. **7.** Драйвер передает набор записей серверу приложений. **8.** Сервер приложений вставляет данные в страницу и передает страницу веб-серверу. **9.** Веб-сервер отправляет подготовленную страницу запросившему ее браузеру.

Для подключения к базе данных сервер приложений использует посредника — драйвер базы данных. Драйвер базы данных представляет собой программный модуль, с помощью которого устанавливается взаимодействие между сервером приложений и базой данных.

После того как драйвер установит соединение, выполняется запрос к базе, в результате чего формируется набор записей. Набор записей представляет собой множество данных, полученных из одной или нескольких таблиц базы данных. Набор записей возвращается серверу приложений, который использует полученные данные для формирования страницы [4].

# **1.3 Преимущества веб-приложений**

Веб-приложение обладает многими преимуществами, в том числе перечисленными ниже.

* Не требует установки на жесткий диск и поэтому не занимает много пространства.
* Не требует обновления, потому что обновляется централизованно.
* Пользоваться можно с любого устройства, на котором есть веб-браузер.
* Не зависит от платформы и операционной системы (ОС): если веб-приложение совместимо с браузером, оно работает.
* Разработчику не требуется создавать клиентские приложения для разных ОС, потому что используется браузер [3].

# **ПРОБЛЕМАТИКА**

В современной России 2022 года кинотеатры находятся в заметном убытке по причине ухода иностранных фильмов с российских площадок. Пройдет какое-то время, прежде чем данный бизнес придется и вовсе закрыть, так как одно лишь местное кино уже не сможет собрать достаточную кассу для поддержания работы сетей и тем более для принесения прибыли.

*«Критический "недостаток репертуара" в ближайшие месяцы повлечет за собой падение выручки кинотеатров в России более чем на 80% и приведет к закрытию минимум 50% кинозалов. Такие оценки содержатся в пояснительной записке к проекту постановления правительства, подготовленному сенаторами из комитета Совета Федерации по экономической политике. [5]»* - DW, Екатерина Венкина от 18 октября 2022 г.

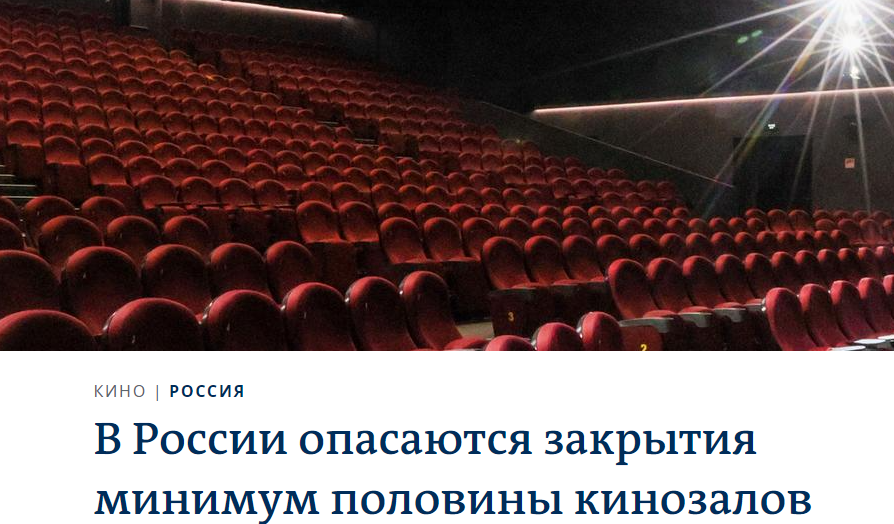


Рисунок 2.1 – Закрытие кинозалов в России

По данным Невафильм Research с начала 2022 года количество кинотеатров в России сократилось на 4,6% (с 2161 до 2062), кинозалов – на 11,3% (с 5704 до 5061). Информация представлена по состоянию на 01.06.2022.

По итогам мая 2022 года неработающими оказались около 770 кинозалов. И открытие залов в новых кинотеатрах, либо повторные открытия залов после временного закрытия или ремонта не способны это компенсировать.

Массовое сокращение числа кинозалов отмечается с марта 2022 года, после отмены проката голливудских фильмов: в марте было не действовало 473 кинозала, в апреле – 148, в мае – 279 [6].

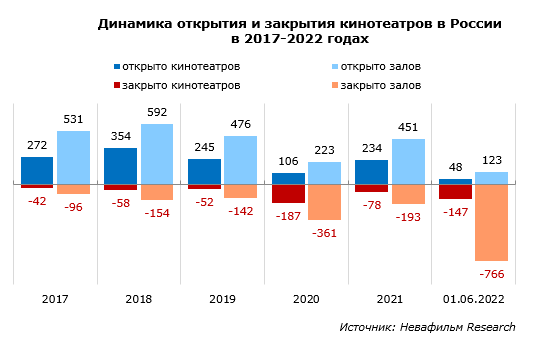


Рисунок 2.2 – Динамика открытия и закрытия кинотеатров в России в 2017-2022 годах

Зарубежные онлайн-кинотеатры также оказались недоступны российскому пользователю, а местные аналоги порой не имеют в своей библиотеке нужного фильма.

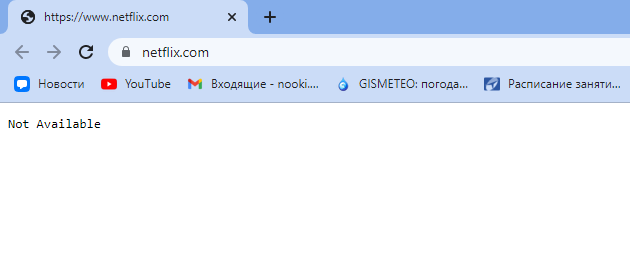


Рисунок 2.3 – Невозможность войти в зарубежные онлайн-кинотеатры

*«В течение полугода после ухода с российского рынка крупных зарубежных киностудий этой весной в онлайн-кинотеатрах страны впервые за историю их существования сократилось количество фильмов и сериалов. Наиболее заметные изменения произошли в библиотеке сервиса Ivi: число наименований на ресурсе сократилось на 42% [7]»* - пишет Forbes 11 октября 2022 года.

Тем не менее, многие все еще хотят наслаждаться вечерними просмотрами фильмов в компании. Поэтому необходимо разработать свою сеть антикинотеатров. Сеансы будут проходить в арендованных помещениях и квартирах, а билет можно будет купить онлайн.

# **ПОСТАНОВКА ЗАДАЧИ**

Как было выяснено, вопрос просмотра кино останется актуальным и после закрытия кинозалов. Для решения этой проблемы был придуман концепт сети антикинотеатров. По задумке заинтересованные данной идеей лица смогут купить билет на местный сеанс просмотра фильмов на дисках, кассетах и с помощью привычного проектора и записи с компьютера/usb-носителя.

Приложение создано для людей, которые скучают по привычной атмосфере кинотеатров. При этом фильм, время и место все еще можно будет выбрать удаленно, а билет купить онлайн.

В стоимость билета входит аренда помещения, затраты на поддержку сайта и некоторый кассовый сбор для продолжения существования проекта. Для брони места необходимо будет перевести половину суммы по реквизитам, высланным после подтверждения покупки. Наличный расчет по факту прибытия может рассматриваться уже после начала работы сервиса и в данной работе рассматриваться не будет.

На сайте должен быть представлен каталог фильмов, разделенных на жанры. Рассматривается возможность разделения по носителям кинофильма (диск, кассета, usb) для любителей, к примеру, старого кассетного кино. При этом классического выбора зала и времени для просмотра фильма не предусмотрено, так как аудитория предполагается не слишком обширной, и собирать зал постоянно не получится. Поэтому используется концепция разового просмотра каждого фильма по вечерам пятниц и на выходных. В дополнение должна быть создана форма для отправки заявки на аренду зала в определенное клиентом время и возможность принести собственный фильм.

Автоматизирование процесса покупки билетов обеспечит большую вероятность сбора полного зала. Яркий интерфейс, соответствующий современным запросам и желаниям, должен привлечь детей и взрослых, заинтересованных в новых (и в то же время знакомых) впечатлениях. База данных заказов не только поможет в организации и автоматизации продажи билетов, но и покажет, в каких фильмах и жанрах наиболее заинтересованы покупатели. Разделение же по жанрам сделает использование сайта еще более простым и удобным.

Стоит отметить, что антикинотеатры уже существуют в России и пользуются некоторой популярностью в своих кругах.

# **ВЫБОР СРЕДСТВ РЕАЛИЗАЦИИ**

Разработка приложения будет вестись в программе **Visual Studio 2019**. К языкам программирования, фреймворкам, а также другими использованным в работе ресурсам относятся HTML, CSS, JavaScript, C#, ASP.NET и Bootstrap.

**HTML** — это стандартный язык разметки, который применяют для создания веб-проектов. Его элементы позволяют отображать стандартные блоки страниц, а также представляют форматированный текст, изображения, таблицы, формы ввода данных и т.д.

Каскадные таблицы стилей (**CSS**) — это язык разметки, который определяет оформление и макет элементов HTML. С помощью CSS задаются шрифты, цвета, стили, расположение отдельных элементов, а также отображение страниц на разных устройствах.

**JavaScript** — это язык программирования, который помогает реализовывать сложное поведение веб-страницы. В большинстве случаев JavaScript используется для создания адаптивных интерактивных элементов для веб-страниц, которые улучшают взаимодействие с пользователем. С помощью JavaScript можно создавать меню, анимацию, видеоплееры, интерактивные карты и даже простые игры в браузере.

Также веб-приложению требуется место для хранения данных, и для этого используется **база данных**. Существует два типа баз данных: реляционные и нереляционные. Реляционные БД хранят структурированные данные, которые обычно представляют объекты реального мира. Нереляционные БД устроены иначе. Речь может идти об объектах с произвольным набором атрибутов. То, что в реляционной БД будет разбито на несколько взаимосвязанных таблиц, в нереляционной может храниться в виде целостной сущности [8].

**C#** — объектно-ориентированный, ориентированный на компоненты язык программирования. Он позволяет разработчикам создавать разные типы безопасных и надежных приложений, выполняющихся в .NET [9].

**ASP.NET** - это один из самых популярных фреймворков для разработки сайтов. Встроенные методы представлений генерируют ясный и соответствующий стандартам код разметки (HTML). Таким образом ASP.NET MVC 5 предлагает мощную систему маршрутизации URL, которая позволяет создавать удобочитаемые URL-адреса [10].

Фреймворк **Bootstrap** – это набор набор CSS и JavaScript файлов. Чтобы его использовать эти файлы необходимо просто подключить к странице. После этого становятся доступны инструменты данного фреймворка: колоночная система (сетка Bootstrap), классы и компоненты [11].

# **ПРОЕКТ ПРЕДЛАГАЕМОГО РЕШЕНИЯ**

# **5.1 Реализация**

В качестве шаблона проекта было выбрано веб-приложение ASP.NET Core 3.1 с типом проверки подлинности «индивидуальные учетные записи». Включение данной функции позволяет использовать встроенную регистрацию, при которой пользователи смогут создавать собственные аккаунты, хранящиеся в базе данных, которую можно посмотреть с помощью обозревателя объектов SQL Server в Visual Studio.

Минус этого способа заключается в том, что все файлы, относящиеся к такой встроенной регистрации, будут скрыты. Чтобы можно было работать с ними, нам необходимо создать новый шаблонный элемент «Удостоверение» и выбрать все предложенные файлы. Окно данного шаблона представлено на рисунке 5.1.1.

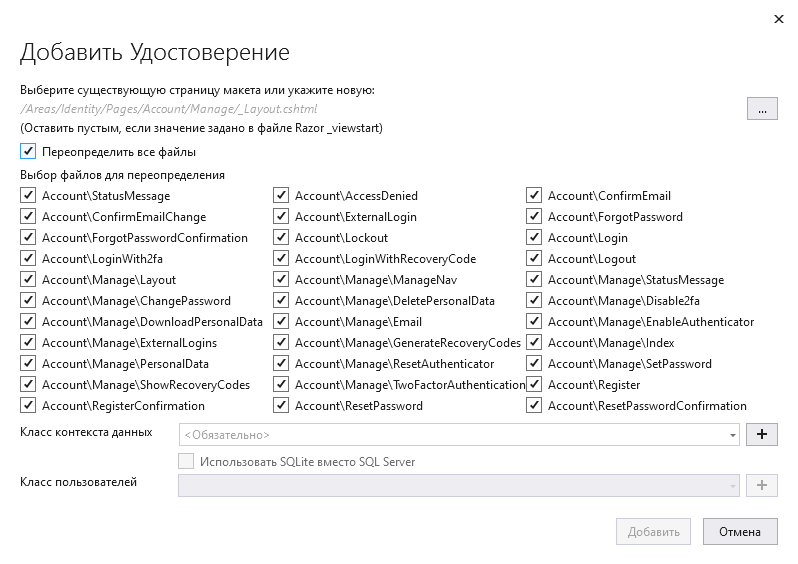


Рисунок 5.1.1 – Окно шаблонного элемента «Удостоверение»

Кроме того, приложение создано с поддержкой MVC. Концепция паттерна MVC предполагает разделение приложения на три компонента:

**Модель (model):** описывает используемые в приложении данные, а также логику, которая связана непосредственно с данными, например, логику валидации данных. Как правило, объекты моделей хранятся в базе данных.

В MVC модели представлены двумя основными типами: модели представлений, которые используются представлениями для отображения и передачи данных, и модели домена, которые описывают логику управления данными.

Модель может содержать данные, хранить логику управления этими данными. В то же время модель не должна содержать логику взаимодействия с пользователем и не должна определять механизм обработки запроса. Кроме того, модель не должна содержать логику отображения данных в представлении.

**Представление (view):** отвечают за визуальную часть или пользовательский интерфейс, нередко html-страница, через который пользователь взаимодействует с приложением. Также представление может содержать логику, связанную с отображением данных. В то же время представление не должно содержать логику обработки запроса пользователя или управления данными.

**Контроллер (controller):** представляет центральный компонент MVC, который обеспечивает связь между пользователем и приложением, представлением и хранилищем данных. Он содержит логику обработки запроса пользователя. Контроллер получает вводимые пользователем данные и обрабатывает их. И в зависимости от результатов обработки отправляет пользователю определенный вывод, например, в виде представления, наполненного данными моделей [12].

По данному принципу все файлы разбиты на папки. Кроме того, существуют также папки с интерфейсами, миграциями, репозиториями и другие. Рассмотрим некоторые из них (файлы из одной папки отличаются лишь названиями переменных и значениями, структура кода у них схожая). Папки для всех файлов представлены на рисунке 5.1.1.

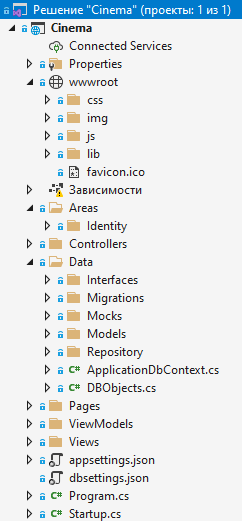


Рисунок 5.1.2 – Папки и файлы

В папке «Models» содержатся модели для жанров, фильмов, заказов, деталей заказов, корзины и предметов в корзине. В таком файле (Order.cs) создадим и регулярные выражения для формы при оформлении заказа.

В папке «interfaces» содержатся интерфейсы для фильмов, заказов и жанров. Интерфейсы содержат функции, позволяющие вытягивать данные из моделей.

В папке «mock» содержатся моки для фильмов и жанров. Моки содержат информацию для баз данных до миграции. В папке «Controllers» содержатся контроллеры для фильмов, заказов, корзины и главной страницы. В папке «Repository» содержатся репозитории для фильмов, заказов и жанров. Они нужны для вытягивания данных из базы данных.

В папке «Views» содержатся представления для фильмов, заказов, корзины, главной страницы, тело и хэддер всех страниц. Для каждой категории файлов создается своя папка, в которой уже хранятся представления.

В папках «Pages» и «Areas» содержатся файлы, созданные автоматически для встроенной регистрации. Для того, чтобы приложение работало слаженно, файлы в данной папке были немного отредактированы (в том числе изменен язык отображения на странице приложения).

Папка «Migrations» содержит миграции, созданные автоматически при добавлении миграций через консоль диспетчера пакетов для создания таблиц в базе данных.

Папка «wwwroot» содержит картинки, иконку, отображаемую в браузере, css- и js-файлы, а также автоматически созданную библиотеку. Css- и js-файлы содержат лишь ссылки на Bootstrap, в котором находится информация о стилях и скриптах.

При запуске программы открывается файл Program, который вызывает файл Startup. Последний содержит подключения, пути и прочее. Важно прописать здесь все нужные подключения и пакеты, так как иначе мы не сможем их использовать.

# **5.2 Результаты программы**

Главная страница представлена на рисунке 5.2.1.

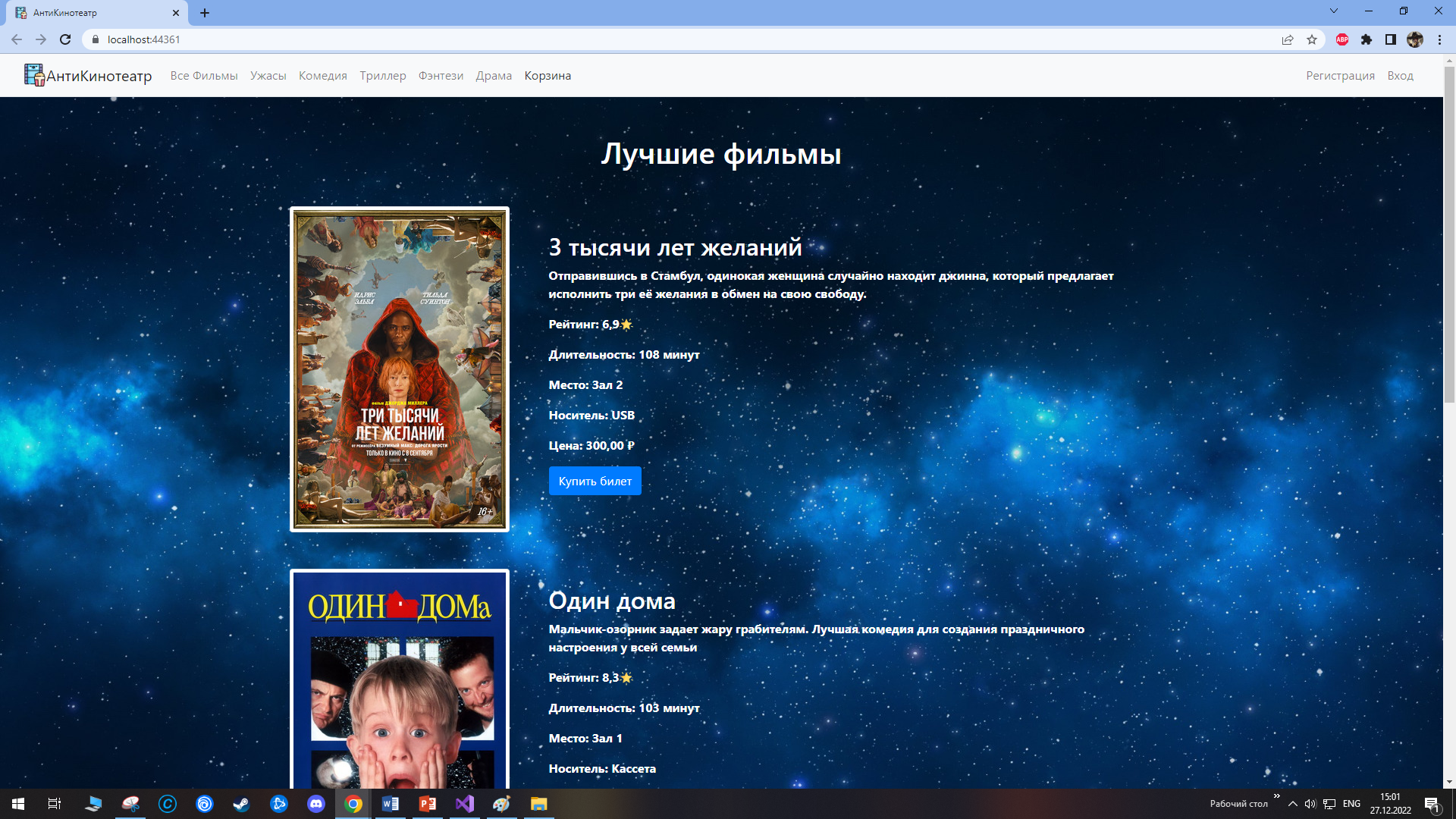


Рисунок 5.2.1 – Главная страница

Страница с каталогом фильмов (нажатие на «Все фильмы») представлена на рисунке 5.2.2.

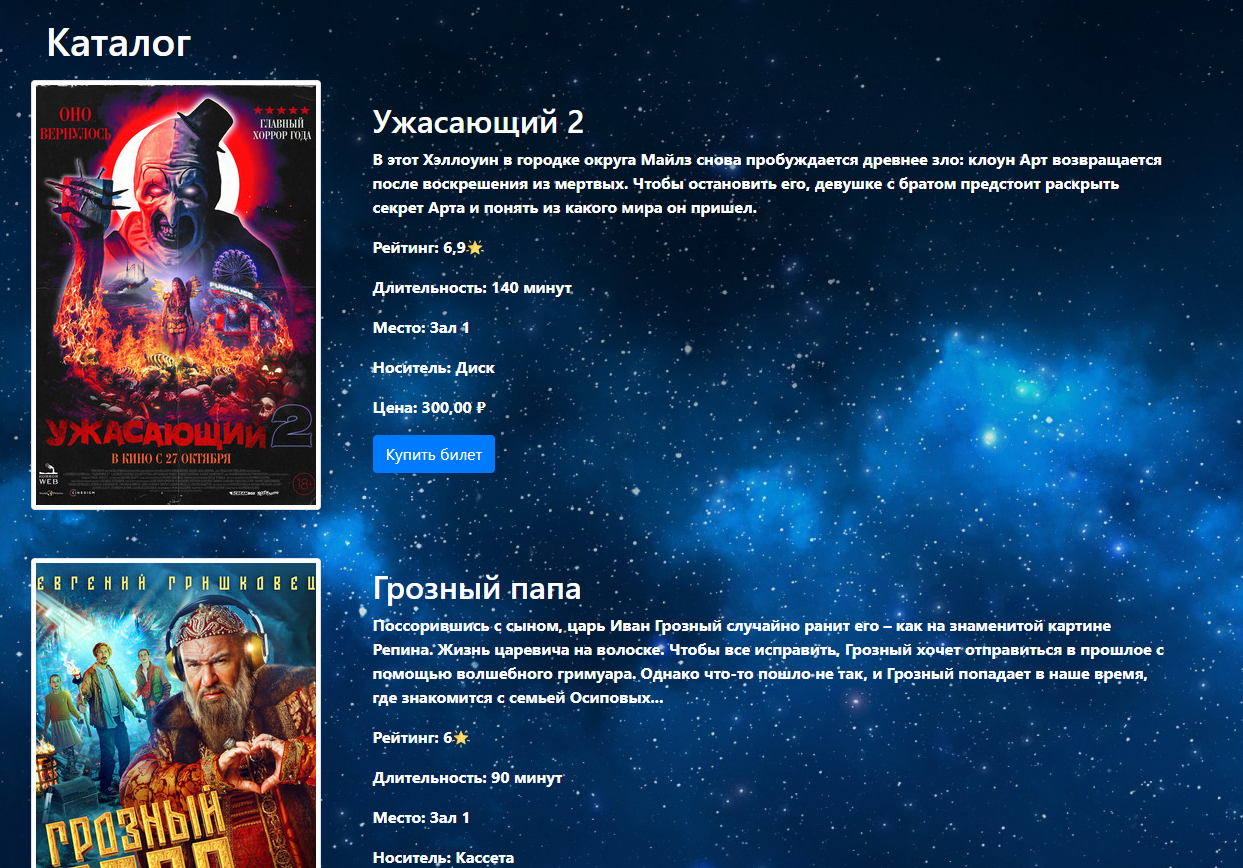


Рисунок 5.2.2 – Каталог фильмов

После добавления фильма в корзину открывается соответствующая страница со списком и ценой игр. Страница представлена на рисунке 5.2.3.

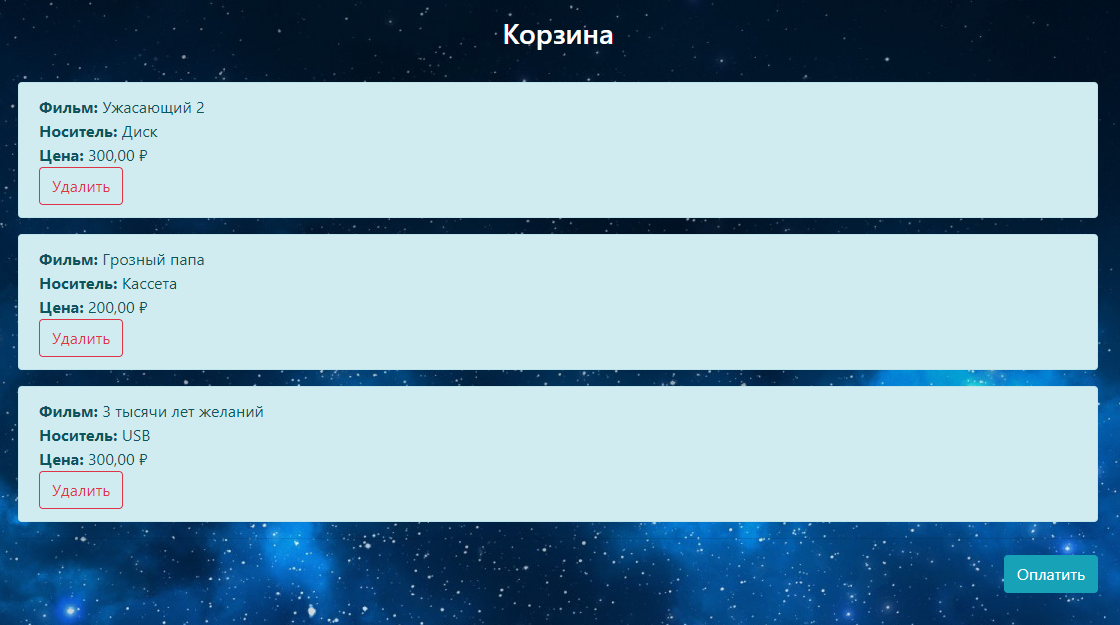


Рисунок 5.2.3 – Корзина

Нажимаем «Оплатить» и приложение перенаправляет нас на форму для оформления заказа. Здесь нужно ввести имя, фамилию, номер телефона, адрес и электронную почту. Страница оформления заказа представлена на рисунке 5.2.4.

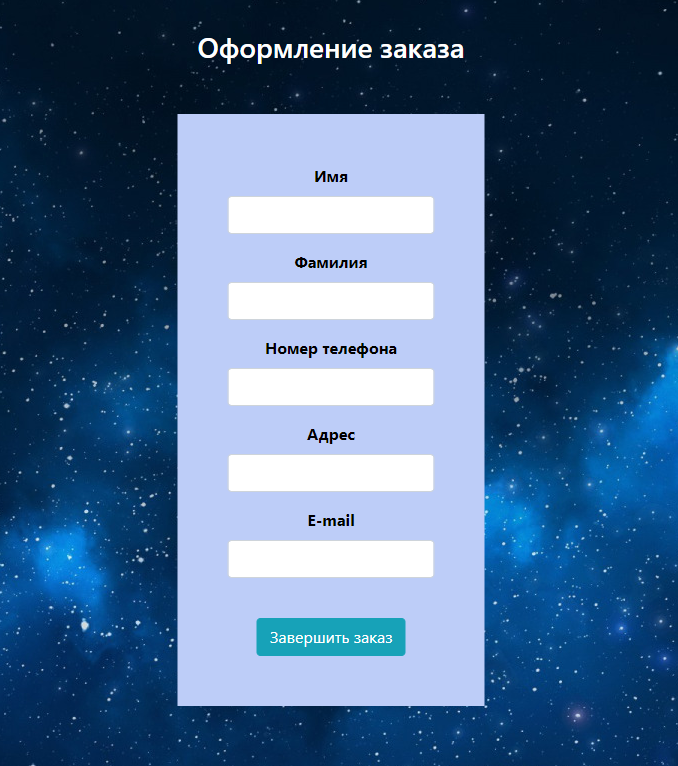


Рисунок 5.2.4 – Оформление заказа

Если мы оставим поля пустыми и попробуем завершить заказ, то увидим, что дальше приложение нас не пускает. Форма с ошибкой представлена на рисунке 5.2.5.

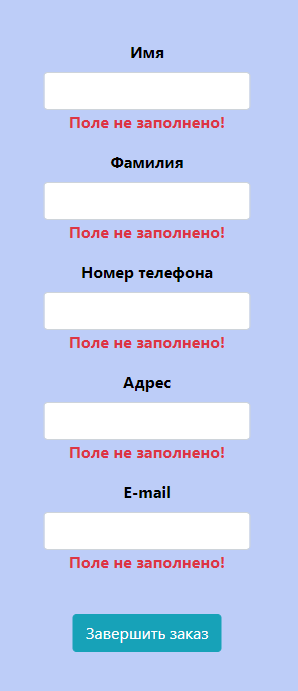


Рисунок 5.2.5 – Ошибка «Поля не заполнены»

Сообщения об ошибке появляются также при неправильном вводе имени, фамилии, номера телефона и электронной почты. Сообщения об ошибках представлены на рисунках 5.2.6 и 5.2.7.



Рисунок 5.2.6 – Валидация электронной почты

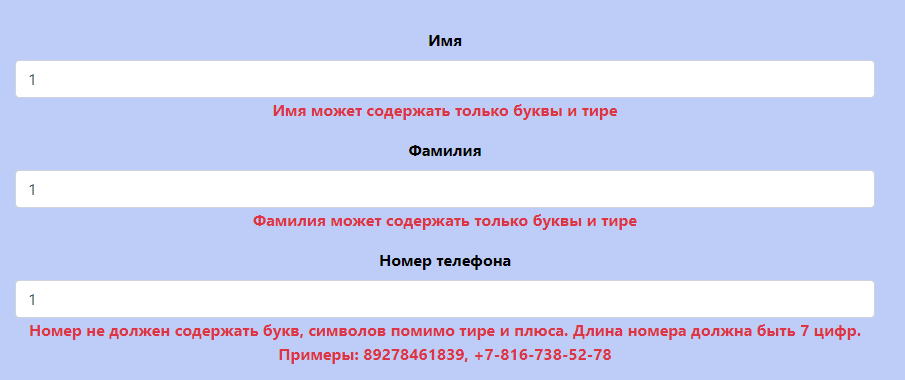


Рисунок 5.2.7– Валидация имени, фамилии и номера телефона

При нажатии завершении заказа с корректным вводом данных, но отсутствием товаров в корзине, мы также увидим ошибку. Сообщение об ошибке представлено на рисунке 5.2.8.

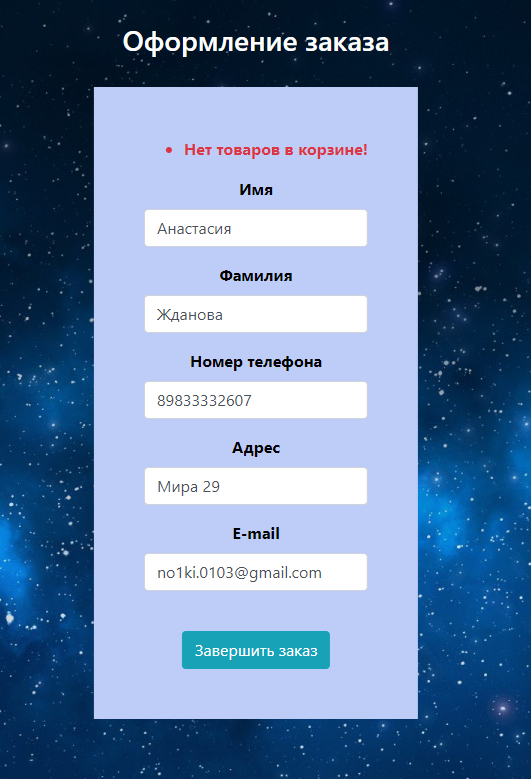


Рисунок 5.2.8 – Ошибка «Отсутствие товаров в корзине»

При нажатии на кнопку регистрации или входа в правом верхнем углу приложение перенаправляет нас на соответствующую страницу. Страницы с регистрацией и входом представлены на рисунках 5.2.9 и 5.2.10.

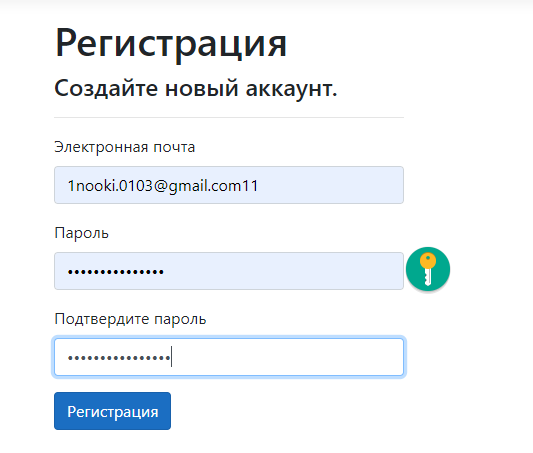


Рисунок 5.2.9 – Регистрация



Рисунок 5.2.10 – Вход

Далее представлены созданные таблицы в базе данных на рисунках 5.2.11-15.

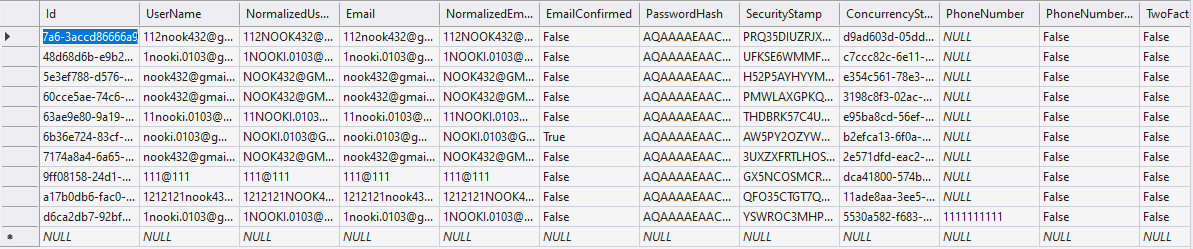


Рисунок 5.2.11 – Таблица «Зарегистрированные пользователи»

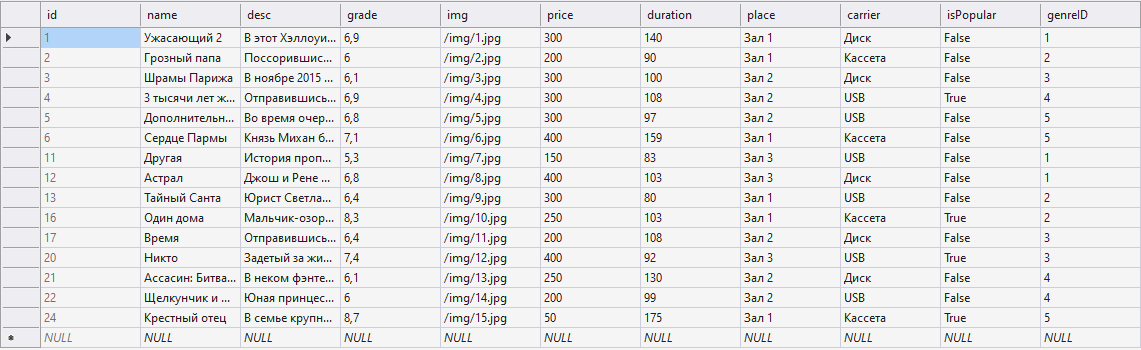


Рисунок 5.2.12 – Таблица «Фильмы из каталога»

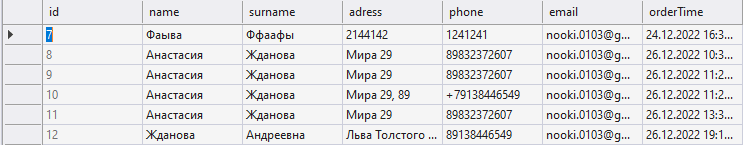


Рисунок 5.2.13 – Таблица «Покупатели»

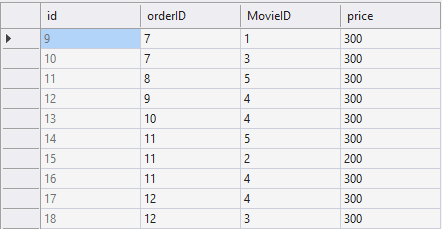


Рисунок 5.2.14 – Таблица «Заказы»

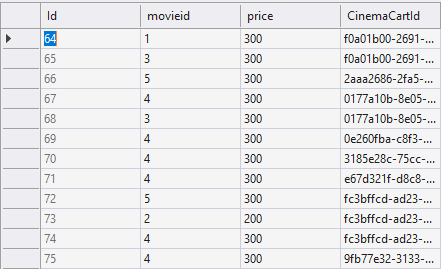


Рисунок 5.2.13 – Таблица «Купленные фильмы»



Рисунок 5.2.14 – Таблица «Жанры»

# **Заключение**

В рамках данной проектной работы было создано веб-приложение «Сайт кинотеатра» на основе веб-фреймворка ASP.NET. Были выполнены поставленные требования для научной учебной работы.

Веб-приложение реализовано с помощью языков HTML, C#, CSS и различных готовых библиотек для них. Также была протестирована работоспособность веб-приложения.

# **Список используемых источников**

1. Автоматизация в сфере производства [электронный ресурс]. – режим доступа: <https://www.nure.info/blog/186-avtomatizaciya-v-sfere-proizvodstva.html> (дата обращения 27.10.2022)
2. Актуальные проблемы автоматизации в современном мире [электронный ресурс]. – режим доступа: <https://cyberleninka.ru/article/n/aktualnye-problemy-avtomatizatsii-v-sovremennom-mire> (дата обращения 27.10.2022)
3. Веб-приложения: виды, архитектура и принципы работы [электронный ресурс]. – режим доступа: <https://highload.today/veb-prilozheniya/#2> (дата обращения 27.10.2022)
4. Общие сведения о веб-приложениях [электронный ресурс]. – режим доступа: <https://helpx.adobe.com/ru/dreamweaver/using/web-applications.html> (дата обращения 27.10.2022)
5. В России опасаются закрытия минимум половины кинозалов [электронный ресурс]. – режим доступа: <https://www.dw.com/ru/v-rossii-opasautsa-zakrytia-minimum-poloviny-kinozalov/a-63468511> (дата обращения 28.10.2022)
6. Рынок кинотеатров в России - итоги мая-2022 [электронный ресурс]. – режим доступа: <https://research.nevafilm.ru/research/research-news/view/cinemamarket-0106202/> (дата обращения 28.10.2022)
7. В библиотеках российских онлайн-кинотеатров впервые сократилось количество контента [электронный ресурс]. – режим доступа: <https://www.forbes.ru/forbeslife/479475-v-bibliotekah-rossijskih-onlajn-kinoteatrov-vpervye-sokratilos-kolicestvo-kontenta> (дата обращения 28.10.2022)
8. Стек технологий для разработки веб-приложений: что важно знать бизнесу [электронный ресурс]. – режим доступа: <https://www.azoft.ru/blog/web-development-stack/> (дата обращения 28.10.2022)
9. Краткий обзор языка C# [электронный ресурс]. – режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/> (дата обращения 28.10.2022)
10. Разработка веб-приложений на C# ASP.NET MVC [электронный ресурс]. – режим доступа: <https://gb.ru/events/521> (дата обращения 28.10.2022)
11. Что такое Bootstrap и зачем он нужен? [электронный ресурс]. – режим доступа: <https://itchief.ru/bootstrap/introduction> (дата обращения 28.10.2022)
12. ASP.NET Core MVC [электронный ресурс]. – режим доступа: <https://metanit.com/sharp/aspnet5/3.1.php> (дата обращения 28.10.2022)