Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В. Ф. УТКИНА»

Кафедра ЭВМ

К защите

Руководитель проекта

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

дата, подпись

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К КУРСОВОМУ ПРОЕКТУ**

по дисциплине

«Администрирование информационных систем»

по теме

«Разработка автоматизированной системы для Web – приложения

«Кинотеатр»

Выполнил студент группы 645

Панин П.Г.

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

дата сдачи на проверку, подпись

Руководитель проекта

доцент кафедры ЭВМ

Свирина А. Г. \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

оценка дата защиты, подпись

Рязань 2019

Содержание

[Введение 3](#_Toc10564914)

[1 Обзор средств и методов автоматизации 4](#_Toc10564915)

[2 Методологии автоматического тестирования 5](#_Toc10564916)

[3 Практическая реализация проекта 7](#_Toc10564917)

[3.1 Разработка прототипа веб-приложения 7](#_Toc10564918)

[3.2 Разработка стратегии тестирования приложения 12](#_Toc10564919)

[3.3 Разработка тест-кейсов 13](#_Toc10564920)

[3.4 Выбор тест-кейсов для автоматизации 29](#_Toc10564921)

[3.5 Разработка автотестов по тест-кейсам 30](#_Toc10564922)

[4 Отчет о тестировании 31](#_Toc10564923)

[Заключение 32](#_Toc10564924)

[Список литературы 33](#_Toc10564925)

[Приложения 34](#_Toc10564926)

[Приложение 1: Листинг разработанных автотестов 34](#_Toc10564927)

# Введение

Целью данного курсового проекта является разработка прототипа Web - приложения в тематике «Кинотеатр» и написание для него стратегии тестирования. В планирование стратегии тестирования входит разработка группы тест-кейсов и автоматизация нескольких из них.

Под автоматизацией подразумевается самостоятельное выполнение компьютером тестовых шагов с помощью некоторого инструмента автоматического тестирования, в нашем случае в качестве такого инструмента будет выступать Selenium, необходимый для автоматизации действий веб-браузера.

# 1 Обзор средств и методов автоматизации

Современное ПО является сложным многофункциональным объектом, и его проверка вручную может занимать достаточно большой отрезок времени. В связи с этим возникла потребность в возникновении средств для автоматизации процессов.

Средства автоматизации – специально разработанные приборы и устройства, функционирующие как в составе аппаратно-программного комплекса, так и автономно, выполняя ряд поставленных задач без вмешательства человека.

В последнее время в качестве средств автоматизации также можно рассматривать отдельные программные продукты, способные выполнять определенные действия над большими объемами данных в соответствие с поставленной задачей внутри какой-либо ЭВМ.

Поэтому одним из наиболее популярных методов автоматизации в современном мире является использование специального программного обеспечения, выполняющего наиболее трудоемкие задачи в кратчайшие сроки без вмешательства человека либо с незначительным его участием в процессе выполнения задач.

# 2 Методологии автоматического тестирования

В настоящее время существует большое количество типов и методов тестирования, которые применяются в зависимости от требований и условий при разработке стратегий тестирования.

Автоматизация тестов является одним из методов сокращения затрачиваемых усилий на проведение тестирования программного продукта, но при возникновении потребности в автоматизации руководствуются правилом, гласящим, что автоматизировать тесты нужно только в том случае, если затраты времени и ресурсов на автоматизацию не превышают затраты на ручное тестирование функционала данного приложения.

Указанное правило соблюдается в одном из направлений тестирования – **регрессионном тестировании**. Данный вид тестирования подразумевает проверку уже созданного функционала при добавлении в программный продукт новых возможностей. Это обусловлено тем, что необходимо проверять работоспособность старого функционала приложения при добавлении нового. Количество таких проверок достаточно велико, поэтому для экономии времени все связанные с этим тесты автоматизируют. Однако, недостатком данного подхода является необходимость переработки автоматизированных тестов при кардинальных изменениях в приложении, затрагивающих ранее написанные функциональные возможности.

Также автоматизация тестов применяется при тестировании интерфейса пользователя, в частности графического. Такой вид тестирования получил название автоматического **GUI-тестирования**. Основная идея данного подхода состоит в проверке корректного функционирования интерфейса приложения. Автоматизация при данном подходе применяется в случае наличия сложных конструкций по выбору разнообразных вариантов деятельности (например, выбор размера, цвета и типа шрифта в MS Word). Для тестирования такого интерфейса потребуется большое количество времени, поэтому здесь к нам приходят на помощь различные инструменты для автоматизации выполняемых действий, например, в курсовом используется инструмент под названием Selenium, который позволяет управлять веб-браузером для выполнения различных манипуляций с веб-приложением.

Часто необходимо проверить, а откроется ли приложение вообще. В данном случае на помощь приходит вид тестирования, который называется **Smoke-тестирование**. Идея данного тестирования заключается в проверке на работоспособность наиболее важного функционала, без которого нет смысла тестировать далее данный программный продукт, а именно запуск приложения, открытие необходимых страниц и форм, появление каких-либо элементов и т.д. На данном этапе можно сразу отловить ошибки, мешающие корректному запуску и стабильной работе приложения. Данный вид тестирования вполне реально проводить вручную, но иногда для экономии времени такие тесты автоматизируют.

В случае, когда необходимо проверить функциональность приложения при использовании его пользователем, прибегают к двум видам тестирования – **тестированию критического пути** и **extended-тестированию**. Тестирование критического пути подразумевает выполнение последовательности действий реального пользователя по достижению какой-либо конкретной цели. Данный способ помогает выявить ошибки в тех направлениях действий, которые будут наиболее часто использоваться, но таким способом невозможно проверить те ошибки, которые могут возникнуть при каких-то непредсказуемых действиях пользователя.

От названого недостатка постарались избавиться в extended - тестировании, здесь имитируется реальная деятельность пользователя в приложении, с учетом всех непредсказуемых действий. Таким образом, данный способ снижает вероятность пропустить какую-либо ошибку, возникающую только в особенно редких случаях.

Во всех перечисленных видах тестирования каждая из проверок дает определенный результат, поэтому решили выделить типы тестирования относительно получаемого результата. Исходя из сказанного выделяют **позитивное** и **негативное** тестирование. Позитивное тестирование направленно на исследование функциональности приложения в поставленных условиях, ожидая при этом благоприятный исход. Если исход оказался отрицательным, то это говорит о наличии ошибок в приложении.

Негативное тестирование наоборот проверяет поведение приложения при неверных исходных данных и прочих неблагоприятных условиях. В данном случае ожидается появление ошибки в приложении, появление различных сообщений относительно неверных данных и др.

Иногда для проверки надежной работы приложения требуется создать для него существенную нагрузку, выраженную большим количеством пользователей, одновременно использующих данный программный продукт. Для достижения таких требований необходимо большая группа людей, которая будет создавать требуемую нагрузку, но это экономически невыгодно и затратно по времени. Применительно к этому случаю используется вид тестирования, который называется **нагрузочным** и относится к группе не функционального тестирования. С помощью специальных программ достигается требуемая нагрузка на тестируемое приложение. Данный вид тоже относится к автоматизированному тестированию.

# 3 Практическая реализация проекта

## 3.1 Разработка прототипа веб-приложения

Разработанное веб-приложение содержит информацию по тематике «Кинотеатр». Приложение содержит 5 основных страниц с переходами между собой и 7 вложенных, а именно:

* Главная:

Содержит общую информацию по заданной тематике, присутствует большое количество ссылок на другие страницы приложения, ссылки на формы с созданием заказа и акциями.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.1):

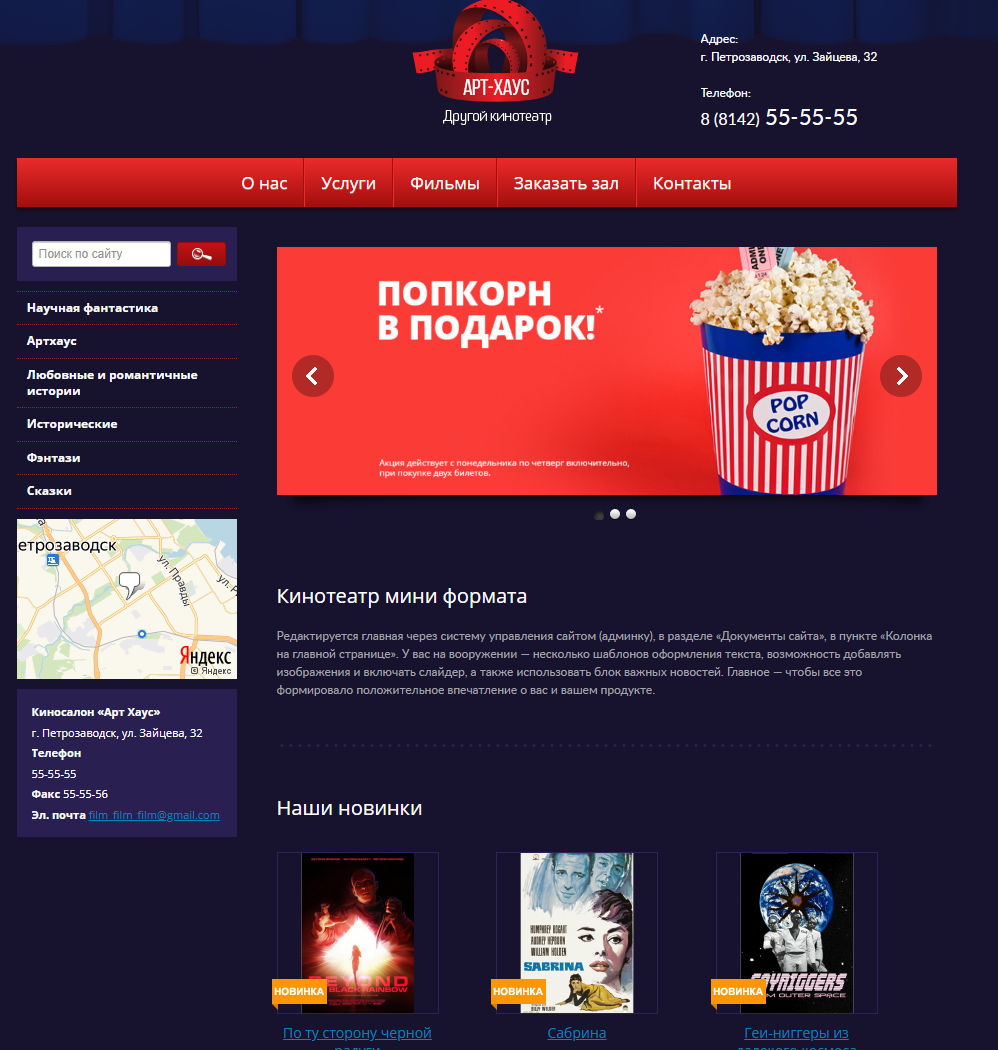


Рисунок 3.1.1 – Главная страница

* Фильмы:

Содержит список жанров фильмов. При выборе жанра перекидывает на страницу с фильмами выбранного жанра.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам и фильмам.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.2):

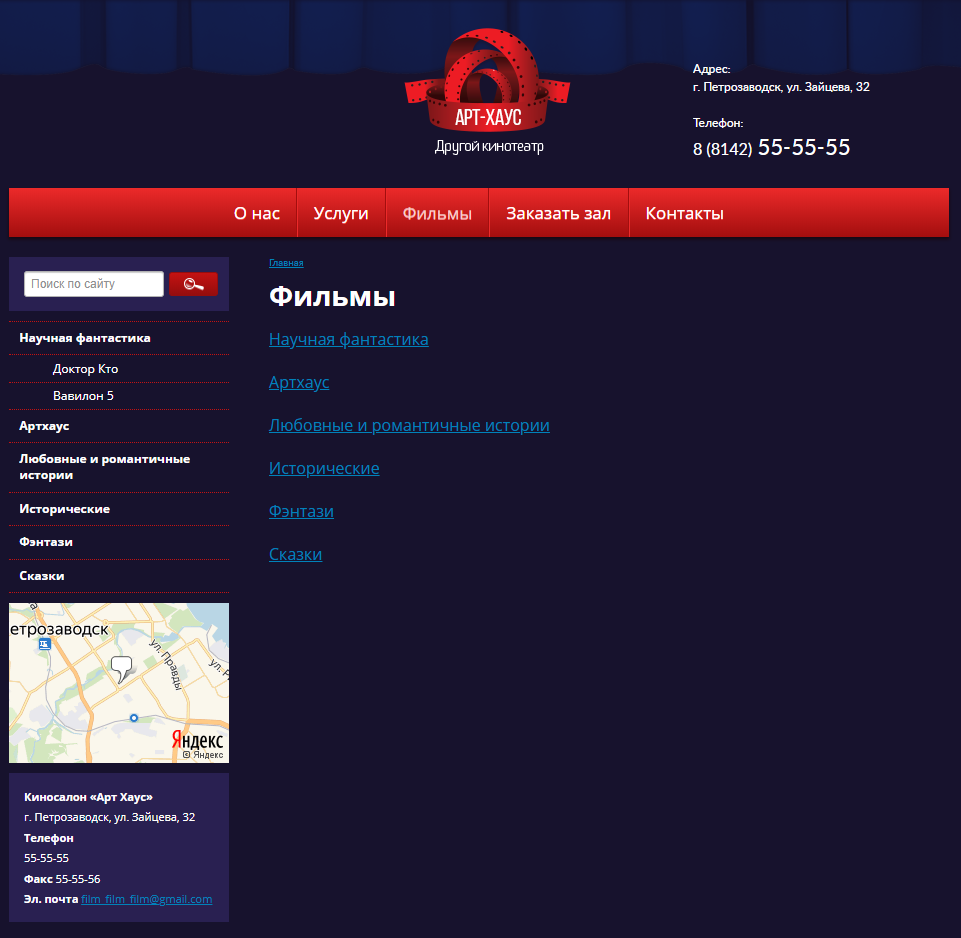


Рисунок 3.1.2 – Страница «Фильмы»

* Заказать зал:

Содержит форму для заявки на заказ зала.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.3):

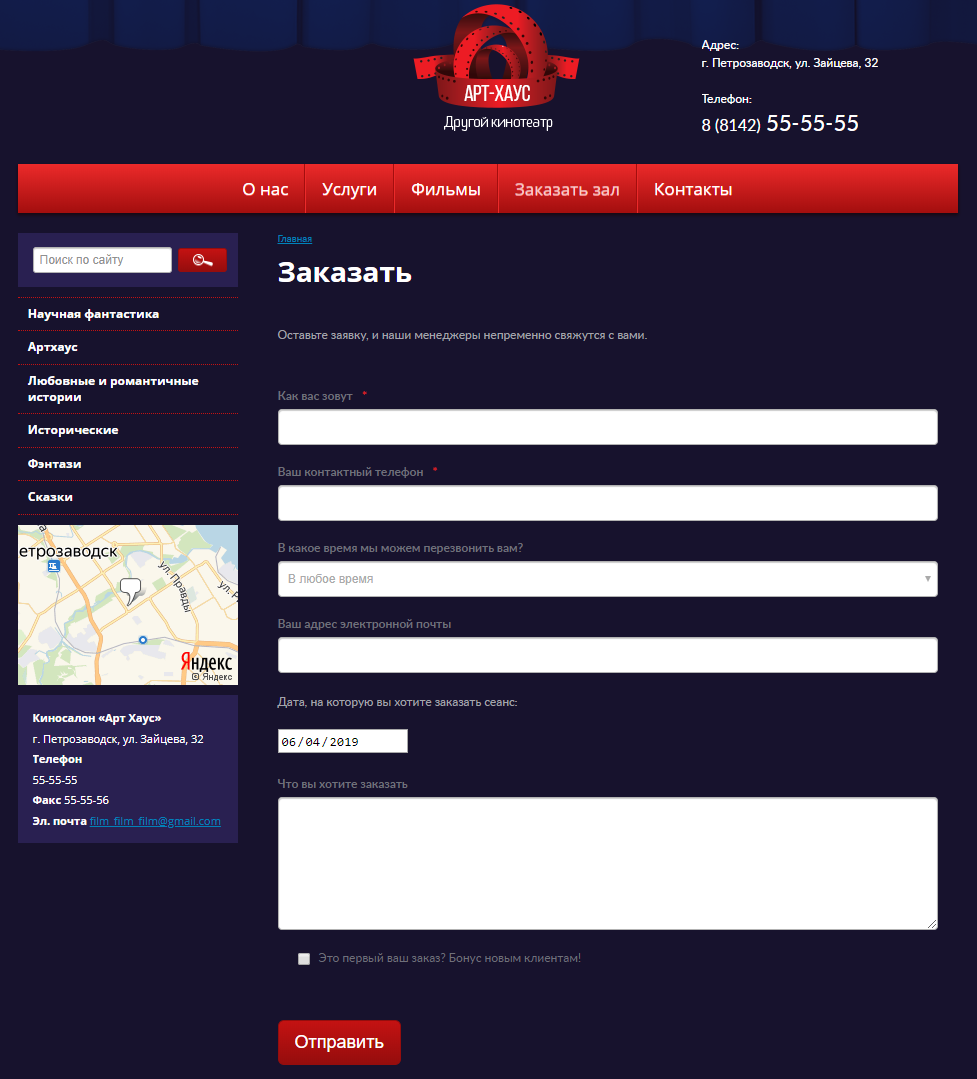


Рисунок 3.1.3 – Страница «Заказать зал»

* Контакты:

Содержит основную контактную информацию о кинотеатре, удобную интерактивную карту города.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.4):

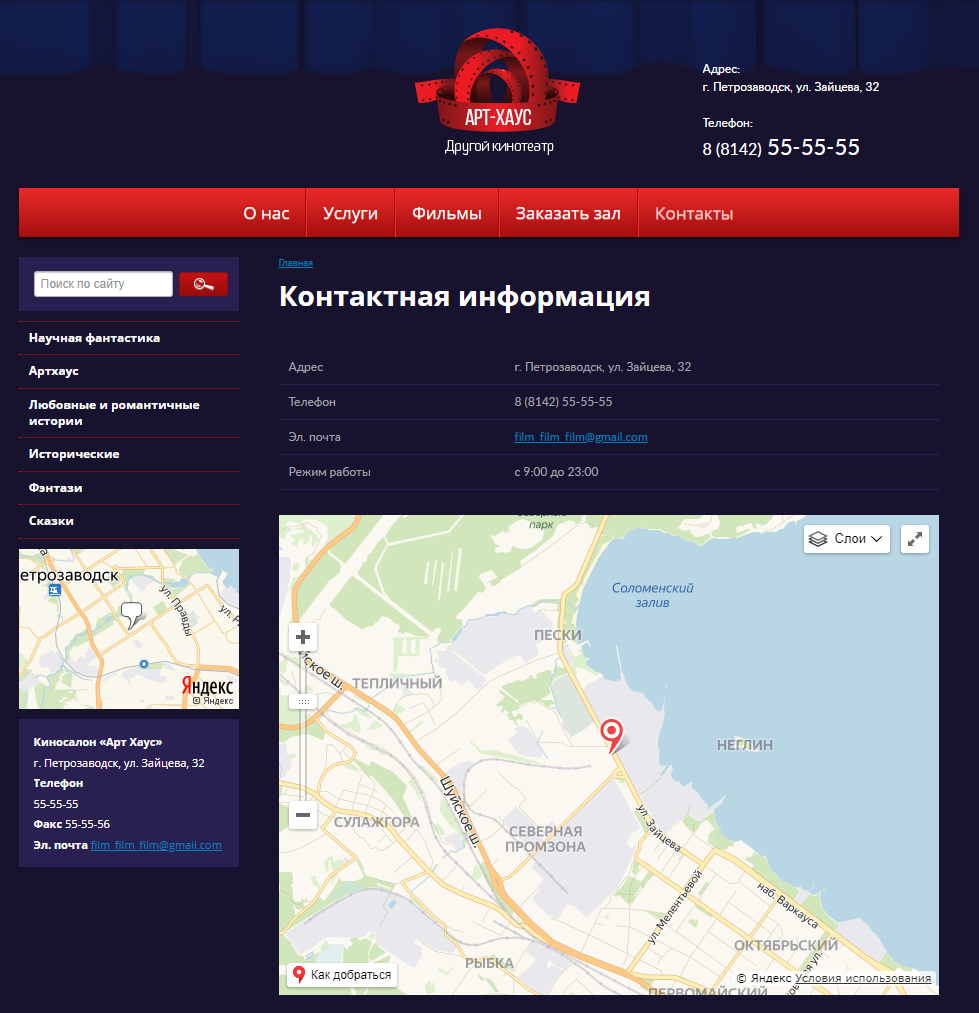


Рисунок 3.1.4 – Страница «Контакты»

Данная страница также содержит форму для отправки вопроса менеджеру (рисунок 3.1.5):

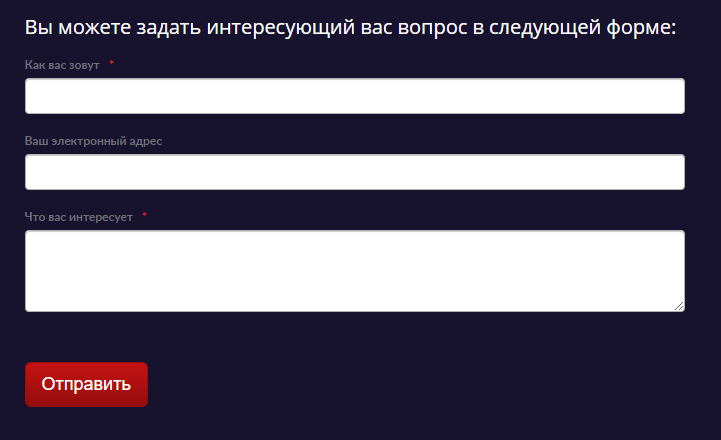


Рисунок 3.1.5 – Форма для вопроса на странице «Контакты»

* Реквизиты:

Содержит основные расчётные данные организации, банковские реквизиты.

Также присутствует список в левой части страницы с быстрой навигацией по разделу «О нас» и жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.6):

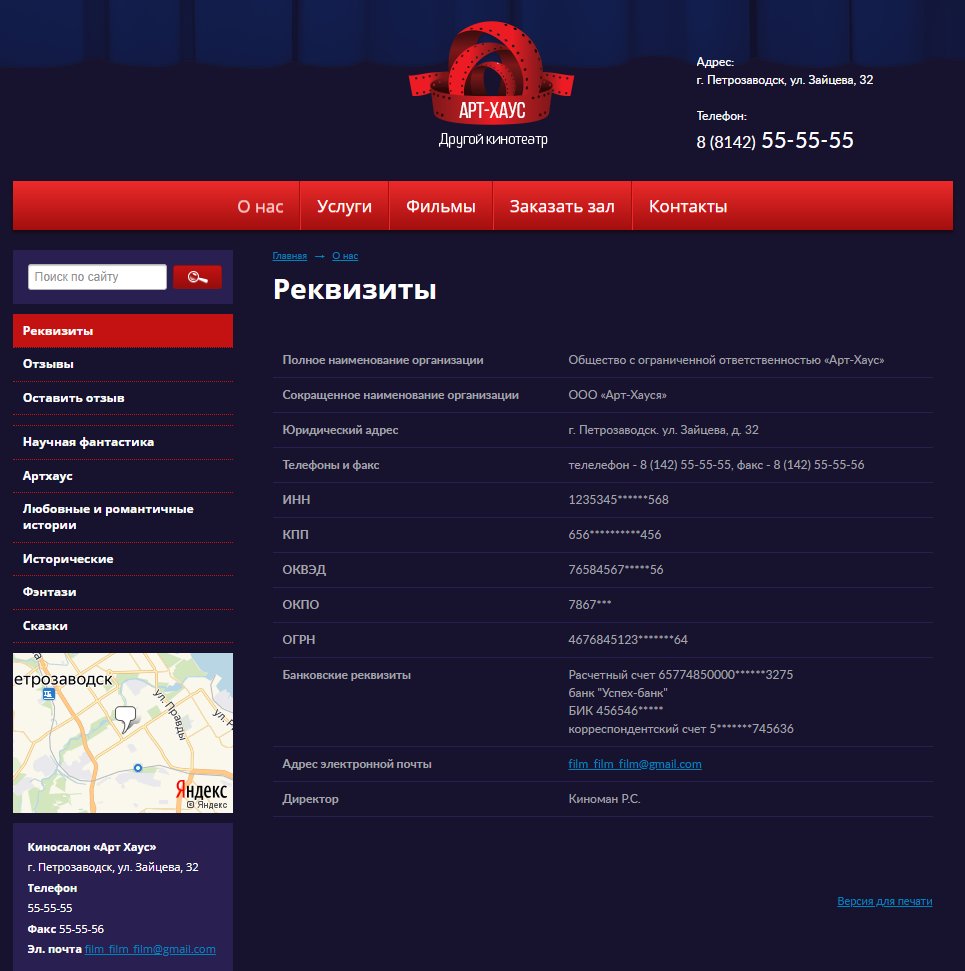


Рисунок 3.1.6 – страница «Реквизиты»

* Отзывы:

Содержит информацию об отзывах и сами отзывы, предложение написать новый.

Также присутствует список в левой части страницы с быстрой навигацией по разделу «О нас» и жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.7):

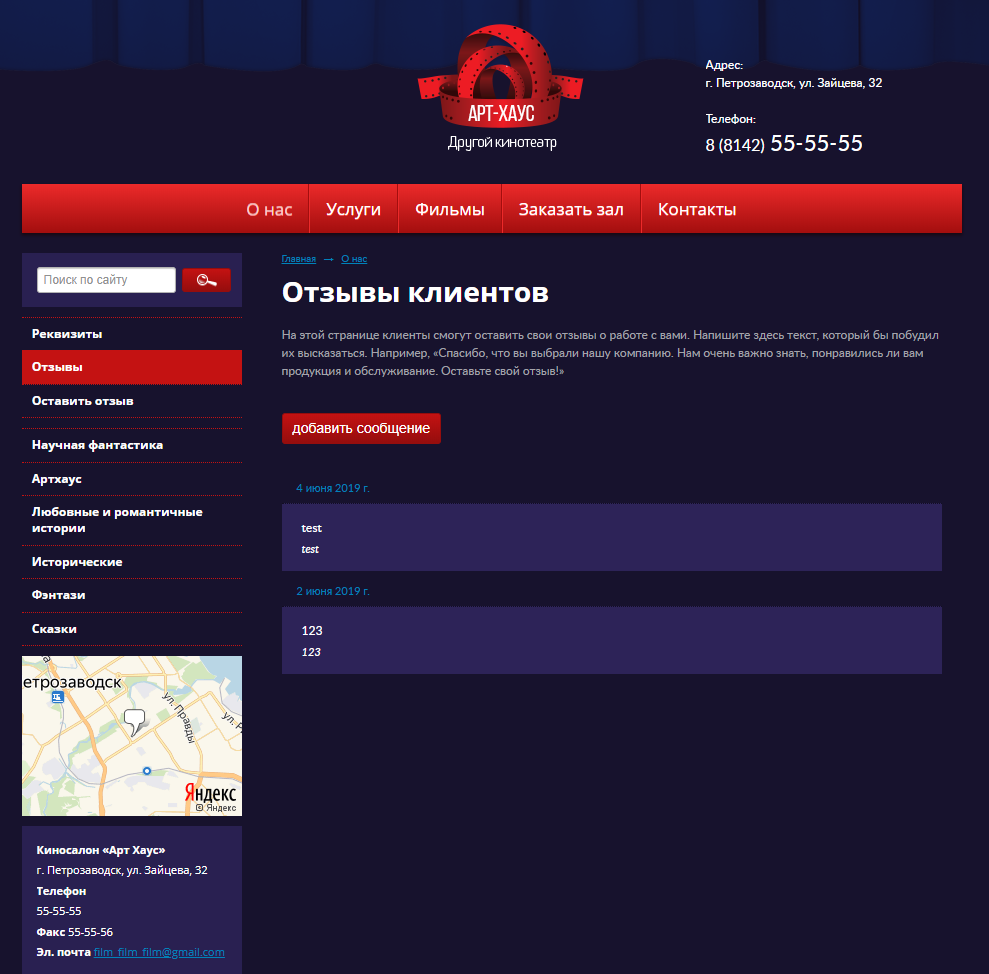


Рисунок 3.1.7 – страница «Отзывы»

* Оставить отзыв:

Содержит форму для написания нового отзыва.

Также присутствует список в левой части страницы с быстрой навигацией по разделу «О нас» и жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.8):

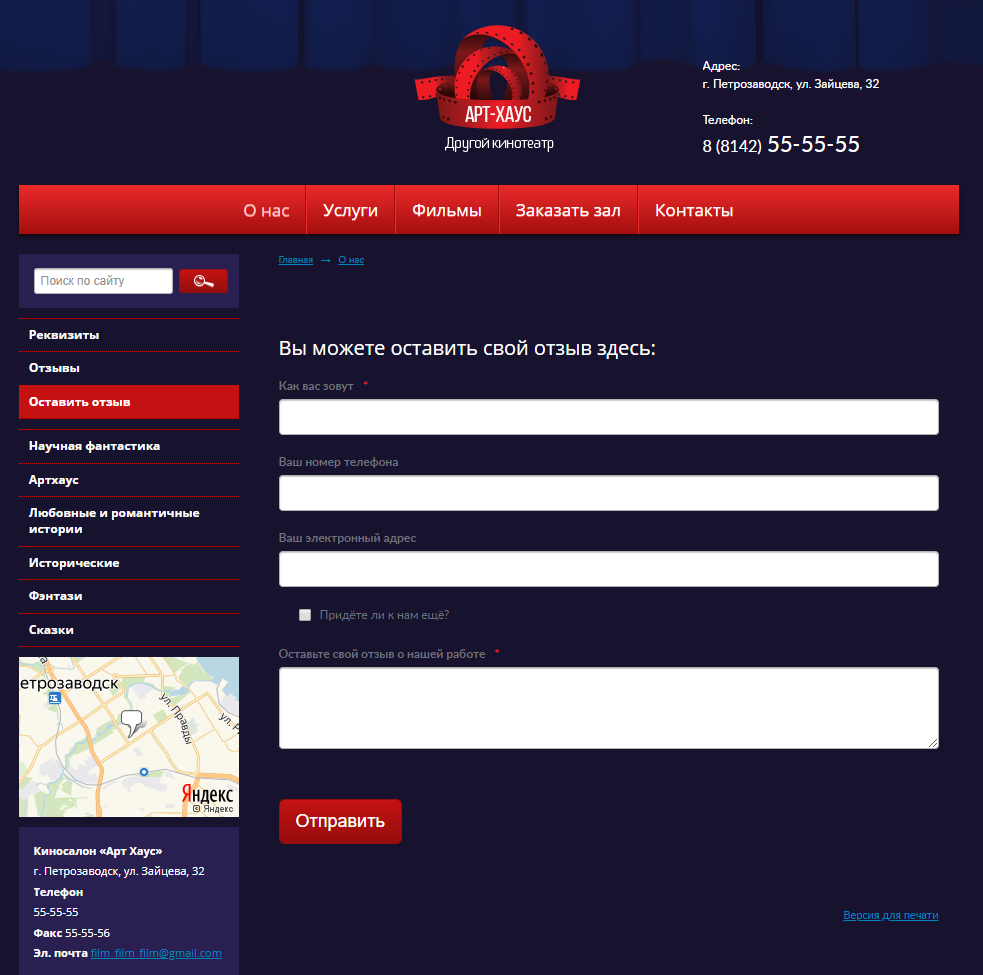


Рисунок 3.1.8 – страница «Оставить отзыв»

* Пример страницы с фильмами выбранного жанра:

Содержит список фильмов определённого жанра, имеется возможность добавить билет в корзину или перейти на странницу с описанием фильма.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.9):

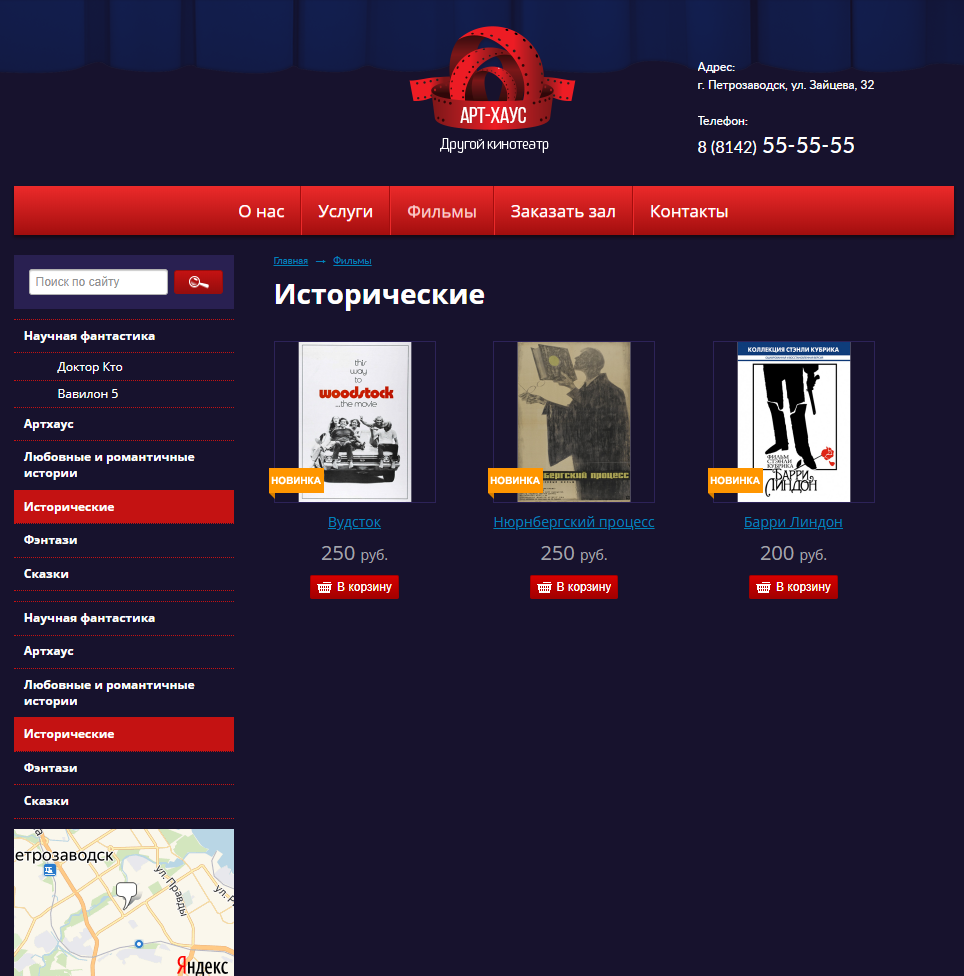


Рисунок 3.1.9 - Пример страницы с фильмами выбранного жанра

* Пример страницы с фильмом:

Содержит описание фильма, цену его билета и информацию об отзывах на этот фильм.

Также присутствует список в левой части страницы с быстрой навигацией по жанрам фильмов.

Ниже этого списка расположена карта города и краткая контактная информация. Данная страница представлена далее (рисунок 3.1.9):

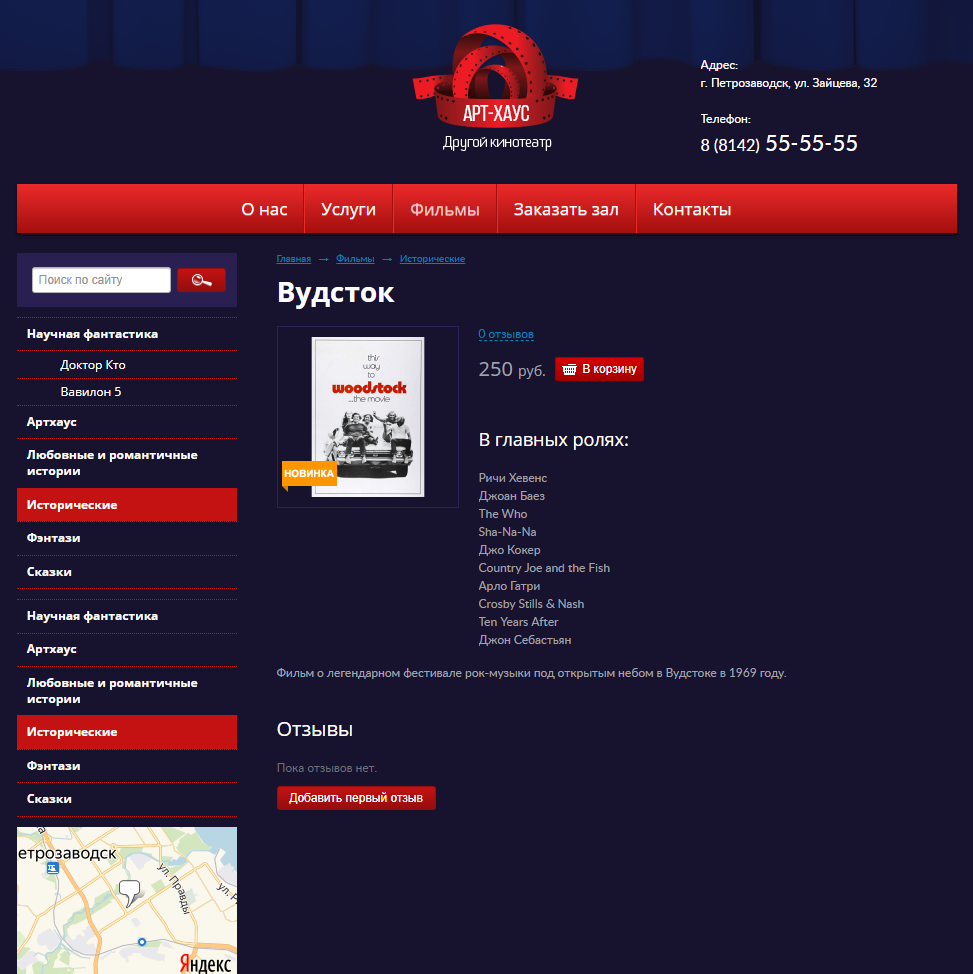


Рисунок 3.1.9 - Пример страницы с фильмом

Большинство реакций форм на отправку, функционала кнопок и реакции элементов форм на различные события реализованы с помощью средств языка JavaScript.

Также предусмотрено сохранение данных, вводимых в формы, с помощью специальной спроектированной серверной независимой части, которая при каждой отправке заполненной формы собирает данные и отправляет на E-Mail адрес администратора, который может передаваться параметрически. Формат, в котором передаются данные, строго структурирован и легко поддаётся дальнейшей обработке (фильтрации) с целью структурированного извлечения данных.

## 3.2 Разработка стратегии тестирования приложения

Для тестирования разработанного веб-приложения необходимо использовать следующие виды тестирования:

1. Компонентное тестирование:

Оно позволяет протестировать функционал отдельных элементов приложения с помощью набора пользовательских тестов.

1. Smoke-тесты, тесты критического пути и extended тесты;
2. Автоматизированные тесты и ручные;
3. Позитивные и негативные тесты.

Критерии качества:

В качестве критериев качества могут выступать работоспособность web-приложения, отсутствие непредвиденных ошибок, стабильная работа приложения и др.

Инструменты тестирования:

В роли инструментов тестирования будут выступать среда разработки MS Visual Studio 2019 Professional с установленным в неё фреймворком NUnit для написания тестов. Для выполнения манипуляций над элементами страниц веб-приложения необходим инструмент Selenium, который позволяет управлять браузером. В качестве браузера был выбран Google Chrome.

## 3.3 Разработка тест-кейсов

Тест-кейс — это профессиональная документация тестировщика, последовательность действий, направленная на проверку какого-либо функционала, описывающая как прийти к фактическому результату.

Тест-кейс должен помогать тестировщику провести проверку продукта без ознакомления с всей документацией. Написанный один раз, удобный в поддержке тест-кейс сэкономит много времени и сил.

Любой тест-кейс включает в себя следующие атрибуты:

* Уникальный идентификатор тест-кейса — необходим для удобной организации хранения и навигации по нашим тест-наборам.
* Название — основная тема, или идея тест-кейса. Кратное описание его сути.
* Предусловия — описание условий, которые не имеют прямого отношения к проверяемому функционалу, но должны быть выполнены.

Например, оставить комментарий на некотором портале может только зарегистрированный пользователь. Значит для тест-кейса «Создание комментария» будет необходимо выполнение предусловия «пользователь зарегистрирован», и «пользователь авторизован»

* Шаги — описание последовательности действий, которая должна привести к ожидаемому результату
* Ожидаемый результат — результат: что ожидается увидеть после выполнения шагов.

Каждый выполненный тест-кейс дает один из трех результатов:

1. Положительный результат, если фактический результат равен ожидаемому результату,
2. Отрицательный результат, если фактический результат не равен ожидаемому результату. В этом случае принято считать, что найдена ошибка.
3. Выполнение теста блокировано, если после одного из шагов продолжение теста невозможно. В этом случае так же принято считать, что найдена ошибка.

Одним тест-кейсом проверяется одна конкретная вещь, и для этой вещи должен быть только один ожидаемый результат.

Чего не должно быть в тест-кейсе:

1. Зависимостей от других тест-кейсов:

Связанный тест-кейс всегда может быть удален из-за ненадобности или он может быть изменен, и в этом случае станет непонятно, как исполнить тест-кейс, на который есть ссылки.

Так же из-за зависимости тест-кейсов, может возникнуть ощущение, что тестируемый продукт уже приведет к нужному состоянию благодаря выполнению связанных тест-кейсов;

1. Нечеткой формулировки шагов или ожидаемого результата:

Если описание шагов или ожидаемого результата будет не четким, то это блокирует прохождение тест-кейса;

1. Отсутствия необходимой для прохождения тест-кейса информации:

В тест-кейсе должна быть вся информация, которая необходима для его прохождения. Например, если проверяется окно логина на сайте, значит понадобится логин и пароль, иначе прохождение этого сценария будет невозможно;

1. Излишней детализации:

Также не следует слишком детализировать кейс. Например, если проверяется возможность создания комментария, то не стоит писать в каком угле экрана должно быть окно логина. Избыточная информация только затрудняет прохождение тест-кейса.

Для тестирования разработанного веб-приложения были составлены следующие тест-кейсы в количестве 20 штук:

1. Тест-кейс № 1

Данный тест-кейс представляет собой смоук-тест, проверяющий ввод пароля на тестовом домене и успешный запуск веб-приложения в браузере (таблица 1):

Таблица 1 - Тест-кейс № 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Actions** | **Requirements** | **Expected Results** |
| UI | | | | |
| Проверка переходов между страницами | Кликнуть ЛКМ на ссылку "Контакты" в шапке | Находиться на любой отличной от "Контакты" странице | Откроется страница "Контакты" |
|
|
| Кликнуть ЛКМ на ссылку "Контакты" в шапке | Находиться на странице "Контакты" | Страница "Контакты" обновится |
|
|
| Кликнуть ЛКМ на ссылку "Заказать зал" в шапке | Находиться на любой отличной от "Заказать зал" странице | Откроется страница "Заказать зал" |
|
|
| Кликнуть ЛКМ на ссылку "Заказать зал" в шапке | Находиться на странице "Заказать зал" | Страница "Заказать зал" обновится |
|
|

1. Тест-кейс № 2

Данный тест-кейс представляет собой проверку выпадающих ссылок (таблица 2):

Таблица 2 - Тест-кейс № 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Actions** | **Requirements** | **Expected Results** |
| UI | | | |
| Проверка выпадающих меню | Подвести курсор к пункту меню "Услуги" в шапке | Курсор должен быть не на этой ссылке | Появится выпадающее меню |
|
|
| Увести курсор с пункта меню "Услуги" в шапке | Курсор должен быть на этой ссылке | Выпдающее меню пропадёт |
|
|
| Подвести курсор к пункту меню "Фильмы" в шапке | Курсор должен быть не на этой ссылке | Появится выпадающее меню |
|
|
| Увести курсор с пункта меню "Фильмы" в шапке | Курсор должен быть на этой ссылке | Выпдающее меню пропадёт |
|
|
| Подвести курсор к пункту меню "О нас" в шапке | Курсор должен быть не на этой ссылке | Появится выпадающее меню |
|
|
| Увести курсор с пункта меню "О нас" в шапке | Курсор должен быть на этой ссылке | Выпдающее меню пропадёт |
|
|

1. Тест-кейс № 3

Данный тест-кейс представляет собой проверку системы оформления заказа (таблица 3):

Таблица 3 - Тест-кейс № 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Actions** | **Requirements** | **Expected Results** |
| UI | | | |
| Проверка системы оформления заказа | Кликнуть ЛКМ на ссылку "Заказать зал" в шапке | - | Откроется страница "Заказать зал" |
|
|
| Ввести имя в поле "Как вас зовут" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Ввести телефон в поле "Ваш контактный телефон" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Выбрать значение из выпадающего списка "В какое время мы можем перезвонить вам?" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Ввести адрес в поле "Ваш адрес электронной почты" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Выбрать дату в поле "Дата, на которую вы хотите заказать сеанс" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Заполнить поле "Что вы хотите заказать" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Поставить галочку на чекбоксе "Это первый ваш заказ?" | Находиться на странице "Заказать зал" | Появится галочка на чекбоксе |
|
|
| Кликнуть по кнопке "Отправить" | Ввести данные во всех обязательных формах | Откроется страница с уведомлением об успешном завершении заказа |
|
|

1. Тест-кейс № 4

Данный тест-кейс представляет собой проверку обязательных к заполнению форм на заполнение (таблица 4):

Таблица 4 - Тест-кейс № 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Actions** | **Requirements** | **Expected Results** |
| UI | | | |
| Проверка необходимости ввода имени при оформлении заказа | Кликнуть ЛКМ на ссылку "Заказать зал" в шапке | - | Откроется страница "Заказать зал" |
|
|
| Не вводить имя, поле "Как вас зовут" оставить пустым | Находиться на странице "Заказать зал" | Поле останется пустым |
|
|
| Ввести телефон в поле "Ваш контактный телефон" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Выбрать значение из выпадающего списка "В какое время мы можем перезвонить вам?" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Ввести адрес в поле "Ваш адрес электронной почты" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Выбрать дату в поле "Дата, на которую вы хотите заказать сеанс" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Заполнить поле "Что вы хотите заказать" | Находиться на странице "Заказать зал" | Введённые данные отобразятся в поле |
|
|
| Поставить галочку на чекбоксе "Это первый ваш заказ?" | Находиться на странице "Заказать зал" | Появится галочка на чекбоксе |
|
|
| Кликнуть по кнопке "Отправить" | Ввести данные не во всех обязательных формах | Страница перезагрузится с выводом сообщения об ошибке "Не заполнено обязательное поле" |
|
|

1. Тест-кейс № 5

Данный тест-кейс представляет собой проверку работоспособности системы написания отзывов о кинотеатре (таблица 5):

Таблица 5 - Тест-кейс № 5

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Actions** | **Requirements** | **Expected Results** |
| UI | | | |
| Проверка работоспособности системы написания отзывов о кинотеатре | Кликнуть ЛКМ на ссылку "О нас" в шапке | - | Откроется страница "О нас" |
|
|
| Кликнуть ЛКМ на ссылку "Оставить отзыв" в левой части экрана | Находиться на странице "О нас" | Откроется форма написания отзыва |
|
|
| Ввести имя в поле "Как вас зовут" | Находиться на странице "Оставить отзыв" | Введённые данные отобразятся в поле |
|
|
| Ввести телефон в поле "Ваш номер телефона" | Находиться на странице "Оставить отзыв" | Введённые данные отобразятся в поле |
|
|
| Ввести адрес в поле "Ваш электронный адрес" | Находиться на странице "Оставить отзыв" | Введённые данные отобразятся в поле |
|
|
| Поставить галочку на чекбоксе "Придёте ли к нам ещё?" | Находиться на странице "Оставить отзыв" | Появится галочка на чекбоксе |
|
|
| Заполнить поле "Оставьте свой отзыв о нашей работе" | Находиться на странице "Оставить отзыв" | Введённые данные отобразятся в поле |
|
|
| Кликнуть по кнопке "Отправить" | Ввести данные во всех обязательных формах | Откроется страница с уведомлением об успешной отправке отзыва |
|
|

1. Тест-кейс № 6

Данный тест-кейс представляет собой переход по ссылкам из блока «Техобслуживание» на главной странице на страницу «Техобслуживание» (таблица 6):

Таблица 6 - Тест-кейс № 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 6 | Проверка работы ссылок на главной странице (блок "Техобслуживание"). | Открытый сайт и активная главная страница. | 1. Навести курсор на блок "Техобслуживание" и щелкнуть по иконке с шестерней. | 1. Откроется страница "Техобслуживание". |

Продолжение таблицы 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 2. Вернуться на главную страницу, нажав пункт главного меню "Главная", и щелкнуть в блоке "Техобслуживание" по строке "Читать далее". | 2. Откроется главная страница и снова произойдет переход на страницу "Техобслуживание". |

1. Тест-кейс № 7

Данный тест-кейс представляет собой переходы по ссылкам из блоков «Перечень услуг» и «Ваши вопросы» главной страницы (таблица 7):

Таблица 7 - Тест-кейс № 7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 7 | Проверка работы ссылок на главной странице (блоки "Перечень услуг" и "Ваши вопросы"). | Открытый сайт и активная главная страница. | 1. Прокрутить страницу вниз и навести курсор на блок "Перечень услуг", щелкнуть по иконке со списком. | 1. Откроется страница "Услуги". |
|
|
| 2. Вернуться на главную страницу, нажав пункт главного меню "Главная", и щелкнуть в блоке "Перечень услуг" по строке "Читать далее". | 2. Откроется главная страница и снова произойдет переход на страницу "Услуги". |
| 3. Снова вернувшись на главную страницу, найти блок "Ваши вопросы" и щелкнуть по иконке со знаком вопроса. | 3. После перехода на главную страницу откроется страница "Связь с нами". |
|
|
|
|
| 4. Вернуться на главную страницу, нажав пункт главного меню "Главная", и щелкнуть в блоке "Ваши вопросы" по строке "Читать далее". | 4. Откроется главная страница и затем произойдет переход на страницу "Связь с нами". |

1. Тест-кейс № 8

Данный тест-кейс представляет собой переходы по ссылкам из нижней части главной страницы (таблица 8):

Таблица 8 - Тест-кейс № 8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 8 | Проверка работы ссылок в конце страницы. | Открытый сайт и активная главная страница. | 1. Прокрутить страницу в конец и щелкнуть по ссылке "Главная". | 1. Произойдет переход на текущую страницу. |
| 2. Снова прокрутить страницу в конец и щелкнуть по ссылке "Услуги". | 2. Откроется страница "Услуги" |
| 3. Вернувшись на главную страницу произвести аналогичные действия с ссылкой "Техобслуживание". | 3. После перехода на главную страницу откроется страница "Техобслуживание". |
| 4. Вернувшись на главную страницу проделать те же действия над ссылкой "Связь с нами". | 4. После перехода на главную страницу откроется страница "Связь с нами". |

1. Тест-кейс № 9

Данный тест-кейс представляет собой заполнение формы подписки на новостную рассылку на главной странице (таблица 9):

Таблица 9 - Тест-кейс № 9

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 9 | Проверка работы формы подписки на новостную рассылку на главной странице. | Открытый сайт и активная главная страница. | 1. Прокрутить страницу в конец и ввести в поля "Имя" и "Email" следующие данные: "Александр" и "testemail@mail.ru", соответственно. | 1. Введенные данные появились в текстовых полях. |
| 2. Нажать на кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 2. Произошла отправка формы и появилось уведомление об успешной отправке формы. |

1. Тест-кейс № 10

Данный тест-кейс представляет собой попытку отправки формы с пустыми полями для ввода (таблица 10):

Таблица 10 - Тест-кейс № 10

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 10 | Проверка запрета отправки формы с пустыми полями ввода. | Открытый сайт и активная главная страница. | 1. Прокрутить страницу в конец и нажать кнопку "Отправить" не заполняя текстовые поля. | 1. Кнопка не нажмется. |

1. Тест-кейс № 11

Данный тест-кейс представляет собой переходы на главную страницу с второстепенный путем нажатия по ссылке "Главная >> " (таблица 11):

Таблица 11 - Тест-кейс № 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 11 | Переход по ссылке на второстепенных страницах. | Открытый сайт и активная второстепенная страница. | 1. На странице "Услуги" щелкнуть по ссылке вида "Главная >> ". | 1. Произойдет переход на главную страницу. |
| 2. Перейти на страницу "Техобслуживание" и щелкнуть по ссылке вида "Главная >> ". | 2. После перехода на страницу "Техобслуживание" откроется главная страница. |
| 3. Перейти на страницу "Связь с нами" и щелкнуть по ссылке вида "Главная >> ". | 3. После перехода на страницу "Связь с нами" откроется главная страница. |

1. Тест-кейс № 12

Данный тест-кейс представляет собой заполнение формы заказа услуг на странице «Услуги» (таблица 12):

Таблица 12 - Тест-кейс № 12

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 12 | Заполнение формы заказа услуг. | Открытый сайт с активной страницей "Услуги". | 1. На странице "Услуги" ввести в поля "Имя" и "Mail" следующие данные: "Виктор" и "victor@mail.com", соответсвенно. | 1. Введенные данные появились в текстовых полях. |

Продолжение таблицы 12

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 2. В пункте "Услуга" отметить нужные нам услуги, пусть это будет "Ремонт трансмиссии". | 2. Напротив выбранной услуги появилась галочка. |
| 3. В пункте "Выберите дату" оставить текущую дату либо ввести свою. | 3. В поле даты отображаться выбранная дата. |
| 4. Нажать кнопку "Отправить", чтобы отправить форму. При появлении сообщения об успешной отправке нажать кнопку "Ок". | 4. Форма отправлена, сообщение об успешной отправке формы выведено. |

1. Тест-кейс № 13

Данный тест-кейс представляет собой заполнение формы заказа услуг на странице «Услуги» всеми возможными вариантами (таблица 13):

Таблица 13 - Тест-кейс № 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 13 (Automatic) | Проверка правильной функциональности всех элементов формы заказа услуг. | Открытый сайт с активной страницей "Услуги". | 1. На странице "Услуги" ввести в поля "Имя", "Mail" и "Телефон" следующие данные: "Олег" , "123" и "89521100129", соответственно. | 1. Введенные данные появились в текстовых полях. |
| 2. В пункте "Услуга" отметить нужные нам услуги, пусть это будет "Ремонт трансмиссии". | 2. Напротив выбранной услуги появилась галочка. |
| 3. В пункте "Выберите дату" оставить текущую дату либо ввести свою. | 3. В поле даты отображается выбранная дата. |
| 4. Нажать кнопку "Отправить", чтобы отправить форму. | 4. Кнопка не нажмется, напротив поля "Mail" выведется сообщение об отсутствии символа "@" в введенной строке. |
| 5. Исправить ошибку, введя в поле "Mail" строку "testmail@mail.ru". | 5. Вводимая строка появится в поле с почтой. |
| 6. В пункте "Услуга" снять галочки со всех услуг и попробовать нажать кнопку "Отправить". | 6. В пункте "Услуги" галочек нет, кнопка "Отправить" не нажимается. |

Продолжение таблицы 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 7. Отметить галочкой услугу "Ремонт двигателя" и попытаться нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 7. Произойдет отправка формы и появится сообщение об успешной отправке, страница перезагрузится. |
| 8. Выполнив повторно шаги 1 и 5, отметить галочкой услугу "Ремонт электропроводки" и попытаться нажать кнопку "Оправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 8. После появления вводимых данных в соответствующие текстовые поля произойдет отправка формы, появится сообщение об успешной отправке, и страница перезагрузится. |
| 9. Выполнив повторно шаги 1 и 5, отметить галочками все услуги пункта "Услуги" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 9. После появления вводимых данных в текстовых полях и галочек в пункте "Услуги" произойдет отправка формы и появится сообщение об успешном отправлении, страница перезагрузится. |
| 10. Выполнив шаги 1 и 5, отметить галочкой любую из услуг, затем нажать кнопку "Очистить". | 10. После заполнения формы произойдет её очистка. |

1. Тест-кейс № 14

Данный тест-кейс представляет собой переход по ссылке из текста страницы «Техобслуживание» на страницу «Связь с нами» (таблица 14):

Таблица 14 - Тест-кейс № 14

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 14 | Переход по ссылке из текста на странице "Техобслуживание". | Открытый сайт и ативная страница "Техобслуживание". | 1. Прокрутить страницу "Техобслуживание" до строки с ссылкой в ней "связаться с нами" и нажать на неё. | 1. Произойдет переход на страницу "Связь с нами". |
| 2. Вернуться обратно на страницу "Техобслуживание" с помощью пункта меню "Страницы/Техобслуживание". | 2. Прозойдет переход на страницу "Техобслуживание". |

1. Тест-кейс № 15

Данный тест-кейс представляет собой заполнение формы подписки на новостную рассылку на странице «Техобслуживание» (таблица 15):

Таблица 15 - Тест-кейс № 15

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 15 | Подписка на новостную рассылку на странице "Техобслуживание". | Открытый сайт с активной страницей "Техобслуживание". | 1. Прокрутить страницу "Техобслуживание" до формы с заголовком "Получать новости" и заполнить поля "Имя" и "Email" данными "Роман" и "mymail@google.com" соответсвенно. | 1. В текстовых полях формы появились вводимые данные. |
| 2. Нажать на кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 2. Произайдет отправка формы и появится сообщение об успешной работе. |

1. Тест-кейс № 16

Данный тест-кейс представляет собой переход с главной страницы на страницу «Связь с нами» и заполнение формы связи на ней (таблица 16):

Таблица 16 - Тест-кейс № 16

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 16 (Automatic) | Переход на страницу "Связь с нами" и проверка функциональности формы на ней. | Открытый сайт с активной страницей "Техобслуживание". | 1. Прокрутить страницу "Техобслуживание" до строки с ссылкой в ней "связаться с нами" и нажать на неё. | 1. Произойдет переход на страницу "Связь с нами". |
| 2. Заполнить поля "Имя", "Mail" и "Телефон" данными "Сережа", "sermail@gmail.com" и "88005553535" соответственно. | 2. Вводимые данные появились в текстовых полях. |
| 3. Нажать на кнопку "Отправить". | 3. Кнопка не нажмется, будет выведено сообщение о необходимости выбора темы из списка. |
| 4. Выбрать в пункте "Тема" любую тему из предложенных, например "Вопрос". | 4. В окне выпадающего списка появится выбранный вариант. |

Продолжение таблицы 16

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 5. Нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 5. Произойдет отправка формы и появится сообщение об успешной работе, страница перезагрузится. |

1. Тест-кейс № 17

Данный тест-кейс представляет собой выполнение всех вариантов заполнения формы связи на странице «Связь с нами» (таблица 17):

Таблица 17 - Тест-кейс № 17

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 17 (Automatic) | Проверка правильной функциональности всех элементов формы на странице "Связь с нами". | Открытый сайт и активная страница "Связь с нами". | 1. Заполнить поля "Имя", "Mail" и "Телефон" данными "Сережа", "sermail" и "88005553535" соответственно. | 1. Введенные данные появились в текстовых полях. |
| 2. Выбрать в пункте "Тема" любую тему из предложенных, например "Вопрос". | 2. В окне выпадающего списка появится выбранный вариант. |
| 3. Нажать на кнопку "Отправить". | 3. Кнопка не нажмется, будет выведено сообщение о том, что поле "Mail" должно содержать символ "@". |
| 4. Ввести в поле "Mail" строку "sermail@gmail.com" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 4. Произойдет отправка формы, появится сообщение об успешной отправке, и страница перезагрузится. |
| 5. Повторив шаги 1 и 4, выбрать в пункте "Тема" тему "Запись на техобслуживание" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 5. В окне выпадающего списка появится выбранный вариант, затем произойдет отправка формы, появление сообщения об успешной отправке, и страница перезагрузится. |
| 6. Подобно шагу 5 выбрать в пункте "Тема" тему "Жалобы и предложения" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 6. Произойдет отправка формы и появится сообщение об успешной работе, страница перезагрузится. |

Продолжение таблицы 17

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 7. Аналогично шагу 5 выбрать в пункте "Тема" тему "Проблемы" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 7. Произайдет отправка формы и появится сообщение об успешной работе, страница перезагрузится. |
| 8. Заполнив форму заново, нажать кнопку "Очистить". | 8. Произойдет очистка формы. |

1. Тест-кейс № 18

Данный тест-кейс представляет собой переходы по всем страницам и заполнение формы подписки на новостную рассылку на странице «Связь с нами» (таблица 18):

Таблица 18 - Тест-кейс № 18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 18 (Automatic) | Имитация действий пользователя (переходы по ссылкам и заполнение формы подписки на новостную рассылку). | Открытый сайт с активной страницей на ней. | 1. На главной странице перейти к блоку "Ремонт" и нажать на ссылку "Читать далее". | 1. Откроется страница "Услуги". |
| 2. Прокрутить страницу "Услуги" в конец и нажать в футере страницы на ссылку "Техобслуживание". | 2. Произойдет переход на страницу "Техобслуживание". |
| 3. На странице "Техобслуживание" перейти в конец и щелкнуть по ссылке из текста "связаться с нами". | 3. Произойдет переход на страницу "Связь с нами". |
| 4. Прокрутить страницу в конец и заполнить форму "Получать новости". В поля "Имя" и "Email" ввести "Александр" и "myemail@mail.ru" соответственно. | 4. В текстовых полях появилась вводимая информация. |
| 5. Нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 5. Произойдёт отправка формы и появится сообщение об успешной работе, страница перезагрузится. |

Продолжение таблицы 18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 6. Вернуться на страницу "Главная" щелкнув по ссылке "Главная >>" в начале страницы. | 6. Открылась страница "Главная". |

1. Тест-кейс № 19

Данный тест-кейс представляет собой переходы по всем страницам и заполнение всех форм, включая формы подписки на новостную рассылку на каждой из страниц (таблица 19):

Таблица 19 - Тест-кейс № 19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 19 (Automatic) | Имитация действий пользователя (переходы по страницам и заполнение форм). | Открытый сайт с активной главной страницей. | 1. Прокрутить страницу "Главная" к блоку "Перечень услуг" и щелкнуть по ссылке "Читать далее". | 1. Откроется страница "Услуги". |
|
| 2. На странице "Услуги" заполнить форму заказа услуг. В поля "Имя", "Mail" и "Телефон" ввести "Павел", "pavmail@gmail.com" и "88005553535" соответственно. | 2. В текстовых полях появилась вводимая информация. |
| 3. В блоке "Услуга" выбрать все предложенные варианты. | 3. Напротив выбранных вариантов появились галочки. |
| 4. В поле "Выберите дату" ввести "2019-05-05". В поле "Комментарий" ввести текст осмысленного содержания. | 4. В поле "Выберите дату" появилась вводимая дата, а в поле "Комментарий" появился вводимый текст. |
| 5. Нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 5. Произойдет отправка формы и появится сообщение о успешной отправке, страница перезагрузится. |

Продолжение таблицы 19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 6. На странице "Услуги" заполнить поля "Имя" и "Email" формы подписки на рассылку строками "Павел" и "pavelmail@gmail.com" соответственно. Нажать кнопку "Отправить", после появления сообщения об успешной отправке, нажать кнопку "Ок". | 6. В текстовых полях появилась вводимая информация. Произошла отправка формы, появилось сообщение об успешной отправке, страница перезагрузилась. |
| 7. Перейти на страницу "Связь с нами" путем нажатия ссылки "Связь с нами" на главном меню сайта. | 7. Открылась страница "Связь с нами". |
| 8. На странице "Связь с нами" заполнить текстовые поля формы. В поля "Имя", "Mail" и "Телефон" ввести "Павел", "pavmail@gmail.com" и "88005553535" соответственно. | 8. В текстовых полях формы появилась вводимая информация. |
| 9. Перейти на страницу "Техобслуживание" нажав на ссылку "Страницы/Техобслуживание" в главном меню сайта либо воспользовавшись ссылкой "Техобслуживание" внизу страницы. | 9. Откроется страница "Техобслуживание". |
| 10. Прокрутив до ссылки "Связаться с нами" в тексте, щелкнуть на неё, чтобы перейти на страницу "Связь с нами". | 10. Произойдет переход на страницу "Связь с нами". Форма связи стала пустой. |
| 11. На странице "Связь с нами" заполнить форму связи. В поля "Имя", "Mail" и "Телефон" ввести "Павел", "pavmail@gmail.com" и "88005553535" соответственно. В блоке "Тема" выбрать пункт "Проблемы" | 11. В текстовых полях формы появилась вводимая информация. В поле выпадающего списка появился выбранный вариант |

Продолжение таблицы 19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 12. Нажать кнопку "Очистить", после ввести в "Имя", "Mail" и "Телефон" ввести "Павел", "pavelmail@gmail.com" и "88005553535" соответственно. | 12. После очистки формы в текстовых полях появилась новая вводимая информация. |
| 13. В блоке "Тема" выбрать пункт "Жалобы и предложения". В блоке "Связь с вами" отметить пункт "По email". | 13. В блоке "Тема" появилась выбранная тема, а в блоке "Связь с вами" отмечен пункт "По email". |
| 14. В блоке "Комментарий" ввести текст осмысленного содержания. | 14. В блоке "Комментарий" появился вводимый текст. |
| 15. Нажать кнопку "Отправить" , при появлении сообщения об успешной отправке нажать кнопку "Ок". | 15. Произойдёт отправка формы и появится сообщение об успешной работе, страница перезагрузится. |
| 16. Повторив шаги 11-14, на шаге 12 в блоке "Тема" выбрать пункт "Вопрос" и нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 16. В текстовых полях появилась вводимая информация. Произошла отправка формы, появилось сообщение об успешной отправке, страница перезагрузилась. |
| 17. Проделать шаг 15 заново. В блоке "Тема" выбрать пункт "Запись на техобслуживание". Нажать кнопку "Отправить" , при появлении сообщения об успешной отправке нажать кнопку "Ок". | 17. В текстовых полях появилась вводимая информация. Произошла отправка формы, появилось сообщение об успешной отправке, страница перезагрузилась. |
| 18. Перейти в конец страницы "Связь с нами" и заполнить поля "Имя" и "Email" формы подписки на рассылку строками "Павел" и "pavelmail@gmail.com" соответственно. | 18. Вводимая информация появилась в текстовых полях формы. |
| 19. Нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 19. Произойдёт отправка формы и появится сообщение об успешной работе. |
| 20. Перейти на страницу "Главная" путем нажатия на ссылку "Четыре колеса" в начале страницы. | 20. Произойдет переход на страницу "Главная". |

Продолжение таблицы 19

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | 21. Перейти в конец страницы "Связь с нами" и заполнить поля "Имя" и "Email" формы подписки на рассылку строками "Павел" и "pavelmail@gmail.com" соответственно. Нажать кнопку "Отправить", при появлении сообщения об успешной отправке нажать кнопку "Ок". | 21. Произойдёт отправка формы и появится сообщение об успешной работе. |
| 22. Перейти на страницу "Техобслуживание" кликнув по строке "Читать далее" блока "Техобслуживание". | 22. Произойдет переход на страницу "Техобслуживание". |
| 23. На странице "Услуги" заполнить поля "Имя" и "Email" формы подписки на рассылку строками "Павел" и "pavelmail@gmail.com" соответственно. Нажать кнопку "Отправить", после появления сообщения об успешной отправке, щелкнуть по кнопке "Ок". | 23. В текстовых полях появилась вводимая информация. Произошла отправка формы, появилось сообщение об успешной отправке, страница перезагрузилась. |
| 24. Вернуться на главную страницу, нажав по ссылке "Главная>>" вверху страницы. | 24. Откроется страница "Главная". |

1. Тест-кейс № 20

Данный тест-кейс представляет собой нажатие по ссылке-якорю на главной странице (таблица 20):

Таблица 20 - Тест-кейс № 20

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № Тест-кейса | Название | Необходимые условия | Производимые действия | Ожидаемый результат |
|
| Тест-кейс № 20 | Проверка работы ссылки-якоря на главной странице. | Открытый сайт и активная страница "Главная". | 1. Прокрутить страницу "Главная" в самый низ и нажать по всплывающей в правом нижнем углу ссылке с пиктограммой стрелки. | 1. Страница прокрутится в самый верх. |

## 3.4 Выбор тест-кейсов для автоматизации

В качестве тест-кейсов для автоматизации были выбраны тест-кейсы с номерами 1.1, 1.3, 1.5, 1.6 1.8, 1.9 и 1.10. Данный выбор был сделан на основе трудозатрат и затрат времени на выполнение данных тестов вручную.

При каких-либо мелких изменениях приложения, не касающихся функций, проверяемых в перечисленных тест-кейсах, необходимо будет проверить весь написанный ранее функционал. Затраты времени на регрессионное тестирование функционала, описанного в выбранных тест-кейсах, превышает затраты на написание автоматизированных тестов на перечисленные тест-кейсы. На основании данных рассуждений можно заключить, что автоматизировать данные тест-кейсы будет рациональнее, чем выполнять описанные там шаги вручную.

## 3.5 Разработка автотестов по тест-кейсам

По выбранным тест-кейсам были разработаны 7 автотестов, которые находятся в проекте “AIS\_CW.sln”. Данные автотесты были разработаны с помощью среды программирования MS Visual Studio 2019 Professional и таких инструментов, как фреймворк NUnit и Selenium Web Driver. Тестирование проводилось в браузере Google Chrome. Список пакетов, установленных для данного проекта, представлен на рисунке 3.5.1:

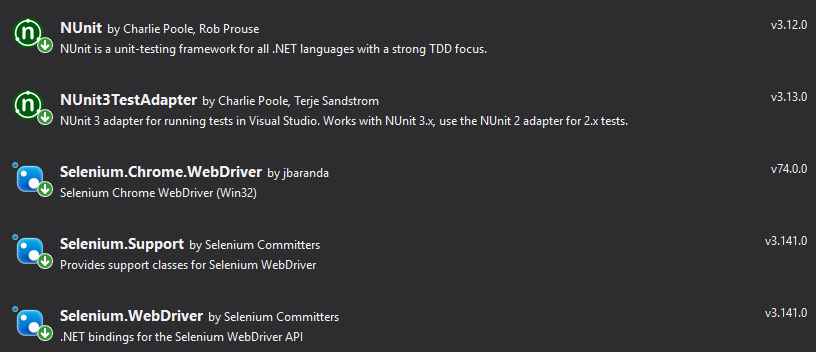


Рисунок 3.5.1 – список установленных пакетов

Код автотестов и веб-приложения (а так же всех материалов, задействованных в разработке данного курсового проекта) размещен на сайте GitHub.com в репозитории по ссылке [<https://github.com/Avendattor/AIS_CW>](https://github.com/Alexander-645/TestWebApplication.git), для удобства открытия которой был сгенерирован следующий QR-код (рисунок 3.5):



Рисунок 3.5 – QR-код с ссылкой на репозиторий с материалами к курсовому проекту.

# 4 Отчет о тестировании

Приложение было протестировано под ОС Microsoft Windows 10 версии Home пакета Single Language с использованием браузера Google Chrome 74.0. Средой для разработки, а так же отладки приложения служила Microsoft Visual Studio 2019 Professional. Смоук-тест успешно был выполнен как вручную, так и с использованием средств автоматизации NUnit 3.0 и Selenium Web Driver. Некоторые тесты критического пути были выполнены автоматически, некоторые – вручную.

На версии последней сборки все автоматизированные тесты были выполнены на 100% (что представлено на рисунке 4.1), в ходе проверки дефектов не обнаружено: приложение работает стабильно, основная функциональность работоспособна.

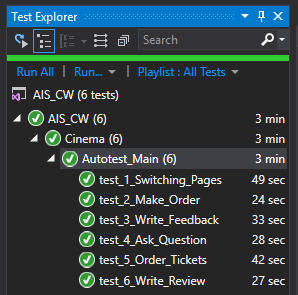
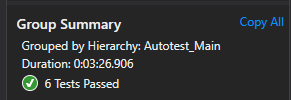
 

Рисунок 4.1 – результаты выполнения автоматизированных тестов

Всего было выполнено порядка 25 проверок, все возникшие во время их выполнения дефекты были устранены.

# Заключение

В ходе курсового проекта было разработано простейшее веб-приложение на тему «Кинотеатр», написаны тест-кейсы для тестирования функциональности приложения, и выполнено тестирование программного продукта разработанными по тест-кейсам автотестами.

Были изучены основы использования инструмента Selenium для управления веб-браузером, а также получены практические навыки написания тестов на языке C# с помощью фреймворка для тестирования NUnit в среде программирования Microsoft Visual Studio Professional.

# Список литературы

1. Свирина А.Г.: Конспект лекций по дисциплине «Администрирование информационных систем»; РГРТУ; 2019г.
2. Документация по Selenium WebDriver [Электронный ресурс]: URL: <https://www.seleniumhq.org/docs/> (дата обращения: 25.05.2019).
3. Руководство по настройке и использованию Selenium [Электронный ресурс]: URL: <https://www.toolsqa.com/selenium-c-sharp/> (дата обращения: 26.05.2019).
4. Microsoft Docs Модульное тестирование кода C# с использованием NUnit и .NET Core [Электронный ресурс]: URL: <https://docs.microsoft.com/ru-ru/dotnet/core/testing/unit-testing-with-nunit> (дата обращения: 26.05.2019).
5. Автоматизированное тестирование [Электронный ресурс]: URL: <https://ru.wikipedia.org/wiki/Автоматизированное_тестирование> (дата обращения: 29.05.2019).
6. Stack Overflow [Электронный ресурс]: URL: <https://stackoverflow.com/> (дата обращения: 29.05.2019).
7. Software Quality Assurance & Testing Stack Exchange [Электронный ресурс]: URL: <https://sqa.stackexchange.com/> (дата обращения: 29.05.2019).
8. ​Хабр [Электронный ресурс]: URL: <https://habr.com/> (дата обращения: 30.05.2019).
9. Oodles Technologies [Электронный ресурс]: URL: <https://www.oodlestechnologies.com/> (дата обращения: 01.06.2019).

# Приложения

## Приложение 1: Листинг разработанных автотестов

**Program.cs:**

using NUnit.Framework;

using OpenQA.Selenium;

using System;

namespace Cinema

{

class Autotest\_Main

{

IWebDriver Browser;

[SetUp]

// Test-case №1.10

public void test\_Initial\_Password()

{

//declaring variable Browser

Browser = new OpenQA.Selenium.Chrome.ChromeDriver();

//Maximizing browser window

Browser.Manage().Window.Maximize();

//opening target site

Browser.Navigate().GoToUrl("https://s22851.nubex.ru/");

//finding password on page

IWebElement Password = Browser.FindElement( By.XPath("/ html / body / div / div[3] / p[2] / strong") );

//ejecting password from element

String PasswordText = Password.Text;

//Searching field to insert password

IWebElement InputPassword = Browser.FindElement( By.Id("password") );

//filling field and sending password

InputPassword.SendKeys( PasswordText + OpenQA.Selenium.Keys.Enter );

//waiting 5 seconds

//System.Threading.Thread.Sleep(5000);

}

[Test]

// Test-case №1.1

public void test\_1\_Switching\_Pages()

{

//Finding element with Contacts link

IWebElement LinkContacts = Browser.FindElement( By.PartialLinkText("Контакты") ) ;

//Clicking on element with Contacts link

LinkContacts.Click();

//waiting 5 seconds

//System.Threading.Thread.Sleep(5000);

//Finding element with Contacts link

LinkContacts = Browser.FindElement(By.PartialLinkText("Контакты"));

//Clicking on element with Order link

LinkContacts.Click();

//waiting 3 seconds

//System.Threading.Thread.Sleep(3000);

//Finding element with Order link

IWebElement LinkOrder = Browser.FindElement(By.PartialLinkText("Заказать зал"));

//Clicking on element with Order link

LinkOrder.Click();

//waiting 5 seconds

//System.Threading.Thread.Sleep(5000);

//Finding element with Order link

LinkOrder = Browser.FindElement(By.PartialLinkText("Заказать зал"));

//Clicking on element with Order link

LinkOrder.Click();

//Finding element with About link

IWebElement LinkAbout = Browser.FindElement(By.PartialLinkText("О нас"));

//Clicking on element with About link

LinkAbout.Click();

//Finding element with Services link

IWebElement LinkServices = Browser.FindElement(By.PartialLinkText("Услуги"));

//Clicking on element with Services link

LinkServices.Click();

//Finding element with Films link

IWebElement LinkFilms = Browser.FindElement(By.PartialLinkText("Фильмы"));

//Clicking on element with Films link

LinkFilms.Click();

}

[Test]

// Test-case №1.3

public void test\_2\_Make\_Order()

{

//Finding element with Order link

IWebElement LinkOrder = Browser.FindElement(By.PartialLinkText("Заказать зал"));

//Clicking on element with Order link

LinkOrder.Click();

//Searching field to insert FIO

IWebElement FIO = Browser.FindElement( By.Name ("form\_data[1]") );

//filling FIO field

FIO.SendKeys("Big Smoke");

//Searching field to insert Phone

IWebElement Phone = Browser.FindElement(By.Name("form\_data[2]"));

//filling Phone field

Phone.SendKeys("88005553535");

//Searching field to insert Call\_time

IWebElement Call\_time = Browser.FindElement(By.Name("form\_data[7]"));

//clicking Call\_time field

Call\_time.Click();

//Searching field to insert Call\_time\_Variants

IWebElement Call\_time\_Variants = Browser.FindElement(By.CssSelector ("#form\_5108 > form > div.formPlain > div:nth-child(3) > div:nth-child(2) > select > option:nth-child(5)"));

//clicking Call\_time\_Variants field

Call\_time\_Variants.Click();

//Searching field to insert email

IWebElement email = Browser.FindElement(By.Name("form\_data[3]"));

//filling email field

email.SendKeys("order@mail.com");

//Searching field to insert date

IWebElement date = Browser.FindElement(By.Name("calendar"));

//filling date field

date.SendKeys("09092019");

//Searching field to insert comment

IWebElement comment = Browser.FindElement(By.Name("form\_data[5]"));

//filling comment field

comment.SendKeys("I’ll have 2 numbers 9s," + OpenQA.Selenium.Keys.Enter +

"Number 9 large," + OpenQA.Selenium.Keys.Enter +

"Number 6 with extra dip," + OpenQA.Selenium.Keys.Enter +

"Number 7," + OpenQA.Selenium.Keys.Enter +

"2 Number 45," + OpenQA.Selenium.Keys.Enter +

"One with cheese," + OpenQA.Selenium.Keys.Enter +

"And a Large soda");

//Searching checkbox to check

IWebElement checkbox = Browser.FindElement(By.Name("form\_data[6]"));

//checking checkbox

checkbox.Click();

//Searching Submit button

IWebElement Submit = Browser.FindElement(By.Name("btn-send"));

//Clicking Submit button

Submit.Click();

}

[Test]

// Test-case №1.5

public void test\_3\_Write\_Feedback()

{

//Finding element with About link

IWebElement LinkAbout = Browser.FindElement(By.PartialLinkText("О нас"));

//Clicking on element with About link

LinkAbout.Click();

//Finding element with Feedback link

IWebElement LinkFeedback = Browser.FindElement(By.PartialLinkText("Оставить отзыв"));

//Clicking on element with Feedback link

LinkFeedback.Click();

//Searching field to insert FIO

IWebElement FIO = Browser.FindElement(By.Name("form\_data[1]"));

//filling FIO field

FIO.SendKeys("Zubenko Mihail Petrovich");

//Searching field to insert Phone

IWebElement Phone = Browser.FindElement(By.Name("form\_data[2]"));

//filling Phone field

Phone.SendKeys("88005553535");

//Searching field to insert email

IWebElement email = Browser.FindElement(By.Name("form\_data[3]"));

//filling email field

email.SendKeys("feedback@mail.com");

//Searching checkbox to check

IWebElement checkbox = Browser.FindElement(By.Name("form\_data[5]"));

//checking checkbox

checkbox.Click();

//Searching field to insert Feedback

IWebElement Feedback = Browser.FindElement(By.Name("form\_data[4]"));

//filling Feedback field

Feedback.SendKeys("Кладмен \*\*\*\*\*, пришлось покупать билет в кинотеатр, чтобы забрать товар. " + OpenQA.Selenium.Keys.Enter +

"А так кинотеатр неплохой, фильмы классные, попкорн вкусный.");

//Searching Submit button

IWebElement Submit = Browser.FindElement(By.Name("btn-send"));

//Clicking Submit button

Submit.Click();

}

[Test]

// Test-case №1.6

public void test\_4\_Ask\_Question()

{

//Finding element with Contacts link

IWebElement LinkContacts = Browser.FindElement(By.PartialLinkText("Контакты"));

//Clicking on element with Order link

LinkContacts.Click();

//Searching field to insert FIO

IWebElement FIO = Browser.FindElement(By.Name("form\_data[1]"));

//filling FIO field

FIO.SendKeys("Zubenko Mihail Petrovich");

//Searching field to insert email

IWebElement email = Browser.FindElement(By.Name("form\_data[2]"));

//filling email field

email.SendKeys("question@mail.com");

//Searching field to insert comment

IWebElement comment = Browser.FindElement(By.Name("form\_data[3]"));

//filling comment field

comment.SendKeys("I can't find on your site 2 numbers 9s," + OpenQA.Selenium.Keys.Enter +

"Number 9 large," + OpenQA.Selenium.Keys.Enter +

"Number 6 with extra dip," + OpenQA.Selenium.Keys.Enter +

"Number 7," + OpenQA.Selenium.Keys.Enter +

"2 Number 45," + OpenQA.Selenium.Keys.Enter +

"One with cheese," + OpenQA.Selenium.Keys.Enter +

"And a Large soda");

//Searching Send button

IWebElement Send = Browser.FindElement(By.Name("btn-send"));

//Clicking Send button

Send.Click();

}

[Test]

// Test-case №1.9

public void test\_5\_Order\_Tickets()

{

//Finding element with Films link

IWebElement LinkFilms = Browser.FindElement(By.PartialLinkText("Фильмы"));

//Clicking on element with Films link

LinkFilms.Click();

//Finding element with Arthouse link

IWebElement LinkArthouse = Browser.FindElement(By.PartialLinkText("Артхаус"));

//Clicking on element with Arthouse link

LinkArthouse.Click();

//Finding element with BTBR to add to cart

IWebElement LinkBTBRToCart = Browser.FindElement(By.XPath ("/html/body/div[2]/div[2]/div[2]/div/div[3]/div/div[1]/div[2]/div[2]/div[2]/form/button"));

//Clicking on element with BTBR link - adding to cart

LinkBTBRToCart.Click();

//waiting for 2 seconds

System.Threading.Thread.Sleep(2000);

//Finding element with Continue link

IWebElement LinkContinue = Browser.FindElement(By.Id("product\_added\_close"));

//Clicking on element with Continue link

LinkContinue.Click();

//waiting for 2 seconds

System.Threading.Thread.Sleep(2000);

//Finding element with Gays Film link

IWebElement LinkFilmGays = Browser.FindElement(By.PartialLinkText("Геи"));

//Clicking on element with Gays Film link

LinkFilmGays.Click();

//Finding element with Gays to add to cart

IWebElement LinkGaysToCart = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div/div[2]/div[1]/div[3]/div[2]/form/button"));

//Clicking on element with Gays link - adding to cart

LinkGaysToCart.Click();

//waiting for 2 seconds

System.Threading.Thread.Sleep(2000);

//Finding element with Cart link

IWebElement LinkToCart = Browser.FindElement(By.PartialLinkText("Перейти в корзину"));

//Clicking on element to go to the Cart

LinkToCart.Click();

//Finding element with adding Gays tickets link

IWebElement ButtonAddGays = Browser.FindElement(By.XPath("//\*[@id=\"cartEdit\"]/div[1]/table/tbody/tr[3]/td[3]/a[2]"));

//Clicking on element with adding Gays tickets link

ButtonAddGays.Click();

//Clicking on element with adding Gays tickets link

ButtonAddGays.Click();

//Clicking on element with adding Gays tickets link

ButtonAddGays.Click();

//Clicking on element with adding Gays tickets link

ButtonAddGays.Click();

//waiting for 2 seconds

System.Threading.Thread.Sleep(5000);

//Finding element with OrderTickets link

IWebElement ButtonOrderTickets = Browser.FindElement(By.PartialLinkText("Оформить заказ"));

//Clicking on element with OrderTickets link

ButtonOrderTickets.Click();

//Searching field to insert FIO

IWebElement FieldFIO = Browser.FindElement(By.Name("form\_data[1]"));

//filling FIO field

FieldFIO.SendKeys("Zubenko Mihail Petrovich");

//Searching field to insert Phone

IWebElement FieldPhone = Browser.FindElement(By.Name("form\_data[3]"));

//filling Phone field

FieldPhone.SendKeys("88005553535");

//Searching field to insert email

IWebElement FieldEmail = Browser.FindElement(By.Name("form\_data[4]"));

//filling email field

FieldEmail.SendKeys("tickets\_order@mail.com");

//Finding element with Cash RadioButton

IWebElement RadioButtonCash = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div[4]/form/div/div[4]/div[2]/label/input"));

//Clicking on element with Cash RadioButton

RadioButtonCash.Click();

//Finding element with Delivery RadioButton

IWebElement RadioButtonDelivery = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div[4]/form/div/div[5]/div[2]/label/input"));

//Clicking on element with Delivery RadioButton

RadioButtonDelivery.Click();

//Searching field to insert Address

IWebElement Address = Browser.FindElement(By.Name("form\_data[8]"));

//filling Address field

Address.SendKeys("г. Петрозаводск, ул. Зайцева, 32");

//Searching field to insert TicketOrderComment

IWebElement TicketOrderComment = Browser.FindElement(By.Name("form\_data[9]"));

//filling TicketOrderComment field

TicketOrderComment.SendKeys("Побыстрее, пожалуйста");

//Searching Send button

IWebElement ButtonSend = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div[4]/form/button"));

//Clicking Send button

ButtonSend.Click();

}

[Test]

// Test-case №1.8

public void test\_6\_Write\_Review()

{

//Searching field to insert Query

IWebElement InputPassword = Browser.FindElement(By.Id("textGray"));

//filling field and sending Query

InputPassword.SendKeys("Геи" + OpenQA.Selenium.Keys.Enter);

//Finding element with Gays Film link

IWebElement LinkFilmGays = Browser.FindElement(By.PartialLinkText("Геи"));

//Clicking on element with Gays Film link

LinkFilmGays.Click();

//Finding element with ButtonNewReview button link

IWebElement ButtonNewReview = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div/div[2]/div[2]/a[2]"));

//Clicking on element with ButtonNewReview button link

ButtonNewReview.Click();

//Searching field to insert FIO

IWebElement FieldFIO = Browser.FindElement(By.Name("signature"));

//filling FIO field

FieldFIO.SendKeys("Сибас Валентинович Валерьянов");

//Searching field to insert Email

IWebElement FieldEmail = Browser.FindElement(By.Name("email"));

//filling Email field

FieldEmail.SendKeys("FilmReview@mail.com");

//Finding element with one star

IWebElement ButtonOneStar = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div[3]/form/div[3]/div[2]/label[5]"));

//Clicking on element with one star

ButtonOneStar.Click();

//Searching ExpTime > year radio button

IWebElement RadioButtonMoreThanOneYear = Browser.FindElement(By.XPath("/html/body/div[2]/div[2]/div[2]/div/div[3]/form/div[4]/div[2]/div[3]/label/input"));

//Clicking ExpTime > year radio button

RadioButtonMoreThanOneYear.Click();

//Searching field to insert Pros

IWebElement FieldPros = Browser.FindElement(By.Name("positive"));

//filling Pros field

FieldPros.SendKeys("Нет");

//Searching field to insert Cons

IWebElement FieldCons = Browser.FindElement(By.Name("negative"));

//filling Cons field

FieldCons.SendKeys("Тоже нет");

//Finding element with Comment link

IWebElement FieldComment = Browser.FindElement(By.Name("comment"));

//Clicking on element with Comment link

FieldComment.SendKeys("Да я и не смотрел этот фильм, в принципе, " + OpenQA.Selenium.Keys.Enter +

"он мне просто не нравится из-за названия.");

//Searching Send button

IWebElement ButtonAddReview = Browser.FindElement(By.Name("send"));

//Clicking Send button

ButtonAddReview.Click();

}

[TearDown]

public void test\_LAST\_Close()

{

//waiting 3 seconds

System.Threading.Thread.Sleep(3000);

//quitting browser after test

Browser.Quit();

}

}

}