Цель работы: провести тестирование документации по критериям.

Описание тестируемой документации:курсовая работа написана по теме «Информационно-справочная система дилерского магазина автомобилей». Документация описывает основные принципы написанного веб-приложения.

Описание критериев качества тестируемой документации:

1. Полнота;
2. Однозначность;
3. Непротиворечивость;
4. Актуальность;
5. Структурированность;
6. Тестируемость.

Описание и обоснование метода тестирования документации: рецензирование (анализ) документации разными группами специалистов. Анализ позволяет выявить проблемы, с которыми знаком каждый из экспертов своей области. Данный метод не требует больших затрат по времени и ресурсам.

Анализ документации:

1. Полнота — все пункты документации описаны в полном объёме вплоть до описания каждой ошибки и каждого окна;
2. Однозначность — документация является однозначной, нет моментов, которые можно воспринять двояко;
3. Непротиворечивость — противоречий не выявлено;
4. Актуальность — документация актуальна, поскольку новых обновлений для веб-приложения не выходило;
5. Структурированность — в документации имеются разделы с понятными названиями, благодаря чему пользователь способен быстро найти нужную ему информацию;
6. Тестируемость — все описанные в документации составляющие можно протестировать вручную или с помощью специальных инструментов.

Выводы по работе: была протестирована документация, приложенная к веб-приложению. Указанная документация не содержит каких-либо явно выявленных недостатков.

Список использованных источников: методичка, моя курсовая.

Приложение: тестируемая документация

Алгоритмические решения

Сервер в режиме реального времени не прекращает ожидать запросы от клиента. Когда запрос получен, его URL адрес начинает сравниваться с соответствующими ссылками функций контроллеров. В случае, если ни одна из функций не обрабатывает данную ссылку, сервер в качестве ответа высылает ошибку 404. В том случае, если запрос сформирован правильным образом, выбранная функция обращается к интерфейсу, для получения необходимых записей из базы данных. После этого данные в качестве ответа отправляются клиенту.

Клиент в данной программе представляет собой совокупность нескольких HTML файлов, которые отображают основную страницу, страницу редактирования записи, страницу добавления записи и страницу «Об авторе». К каждому из них подключена библиотека bootstrap. Она помогает разработчику не прописывать огромное количество свойств в CSS файле (а, может быть, и в самом HTML файле) для улучшения внешнего вида объектов. Достаточно лишь представить дизайн объекта и присвоить ему соответствующий класс для улучшения его внешнего вида. С помощью bootstrap можно сделать даже простую анимацию.

Описание интерфейса программы

Интерфейс программы содержит шесть основных страниц:

— страница входа

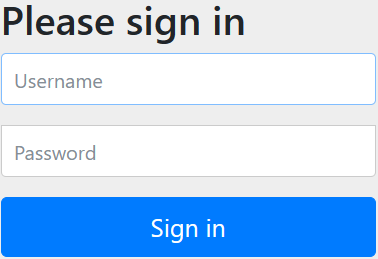
— страница регистрации

— главная страница с таблицей

— страница добавления клиента

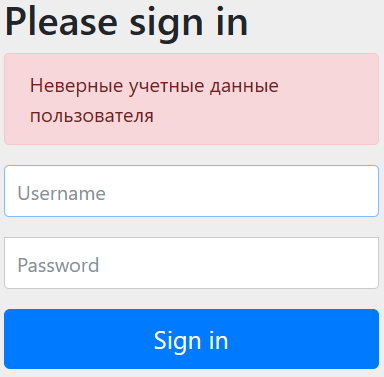
— страница редактирования клиента

— страница «Об авторе»

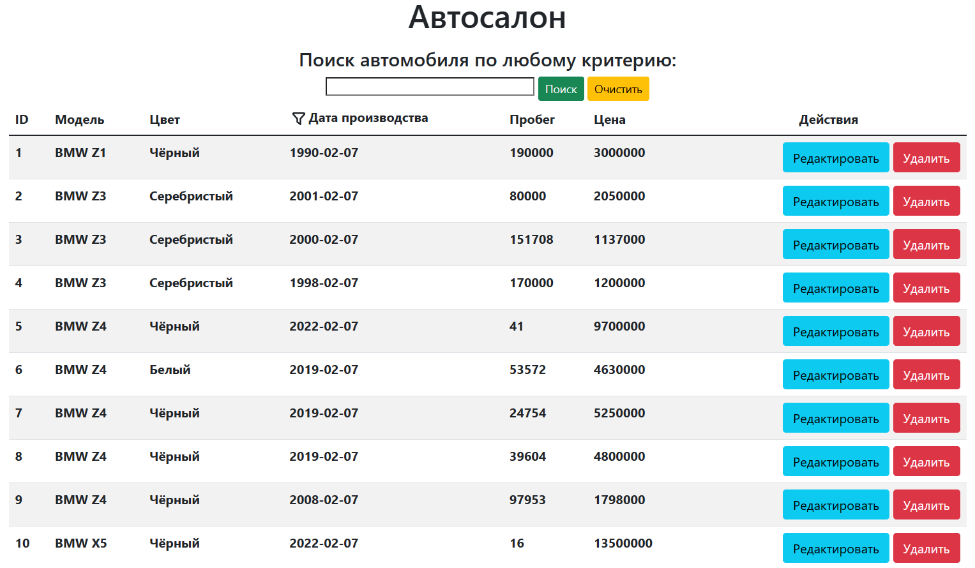


Сразу после нажатия кнопки «Sign in» данные из полей «Username» и «Password» передаются на сервер, где сравниваются с хранящимися там аналогичными данными в файле application.properties. Стоит также упомянуть, что в этом файле хранится и информация для подключения к базе данных, а именно: ссылка на саму базу данных, а также логин и пароль для подключения к ней. Присутствует в этом файле и роль, которая выдаётся пользователю при входе в систему. Так же тут присутствуют настройки кэша.

Как можно заметить, при вводе неверных данных дальнейший доступ к работе с программой не представляется возможным.



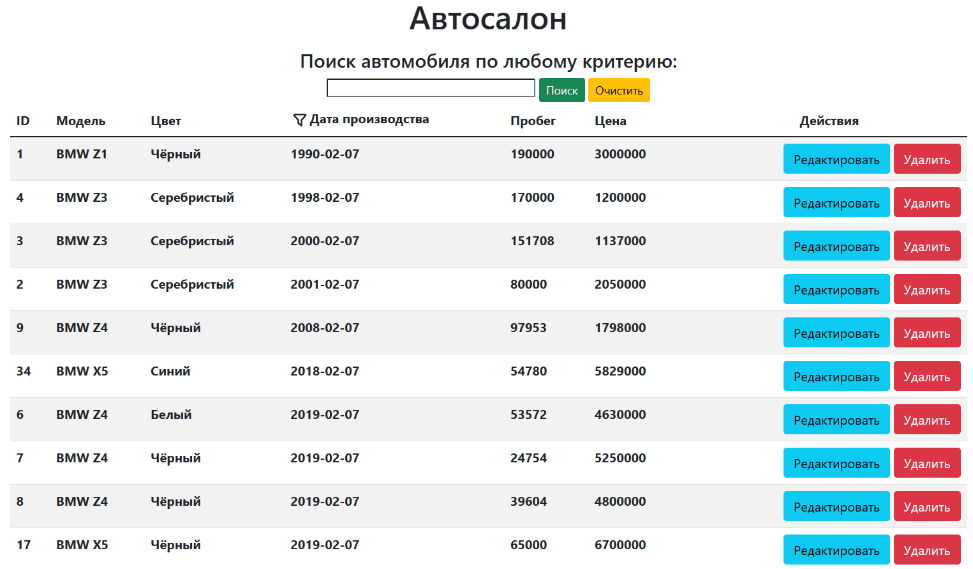
Первое, что увидит пользователь после успешного входа в систему – загруженная база данных.

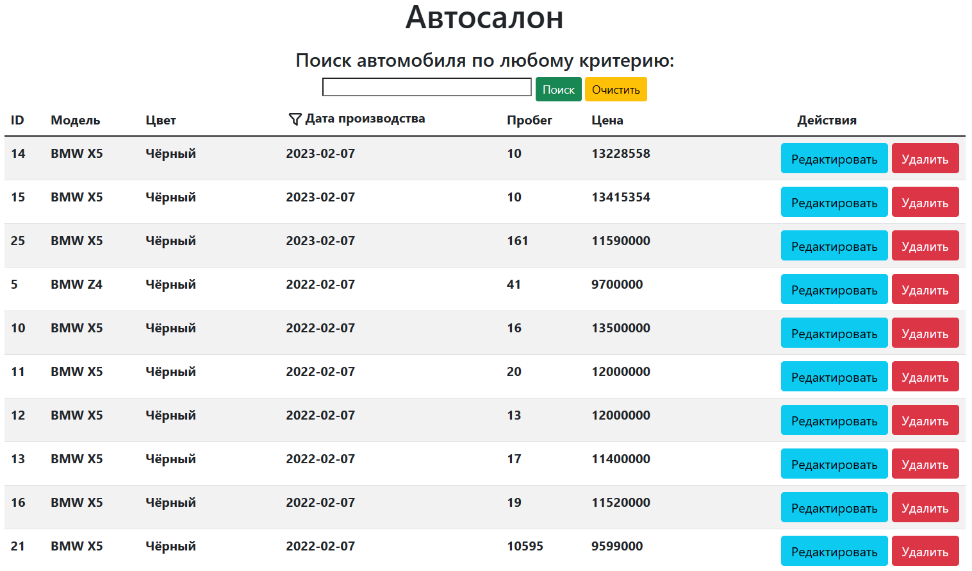


Интерфейс программы интуитивно понятен даже человеку, не сведущему в компьютерах. Для поиска автомобилей по определённому критерию пользователю необходимо ввести последовательность символов в строку поиска и нажать кнопку «Поиск». После этого страница обновится и на ней будут отображаться только подходящие под условия поиска записи из таблицы.

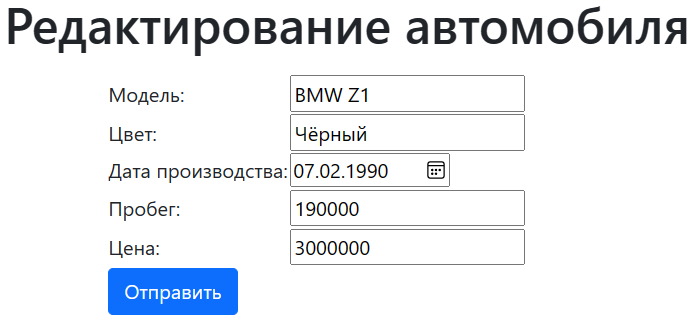


Для того, чтобы вернутся к первоначальному виду страницы, необходимо нажать кнопку «Очистить». Как и в прошлом примере, страница обновится, после чего на ней будут отображаться все записи из таблицы. Записи можно сортировать по возрастанию или убыванию даты производства автомобиля, нажимая на соответствующий значок рядом с колонкой «Дата производства».





Если пользователю необходимо отредактировать определённую запись, он может нажать на соответствующую кнопку, после чего его перекинет на страницу редактирования.



Важный момент! Пользователь не может изменить ID автомобиля или выбрать его при создании новой записи. Это сделано потому, что при создании таблицы в SQL на атрибут ID было наложено ограничение первичного ключа. Первичный ключ – это особенный атрибут в таблице, позволяющий однозначным образом идентифицировать каждую запись в таблице. В связи с этим при создании таблице SQL атрибуту ID было присвоено свойство AUTO\_INCREMENT. Именно благодаря ему ID в каждой новой записи на единицу больше, чем в предыдущей.

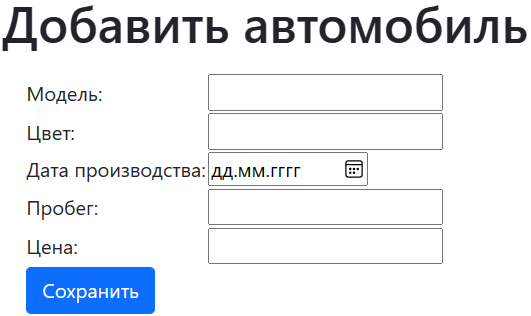
Если пользователь нажмёт на кнопку «Удалить», страница так же обновится, а выбранная запись будет удалена. Замечу, что создать новую запись с тем же ID, который был у удалённой записи, невозможно. В таком случае можно просто пересоздать таблицу.

Чуть ниже таблицы вниманию пользователя представлена кнопка «Добавить автомобиль», которая перекидывает юзера на соответствующую страницу.

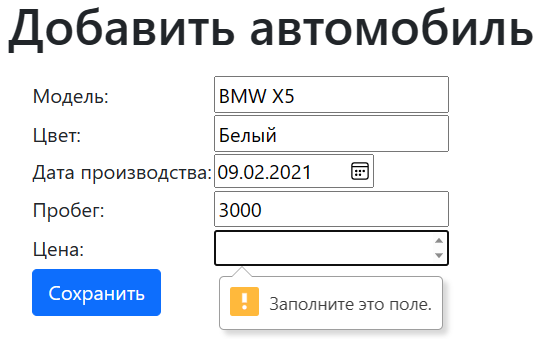


Форма регистрации нового автомобиля в системе выглядит следующим

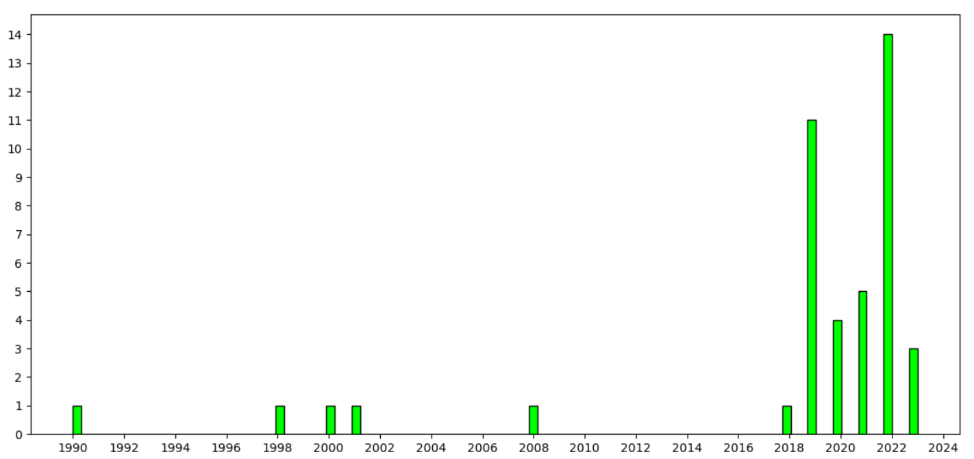
образом:



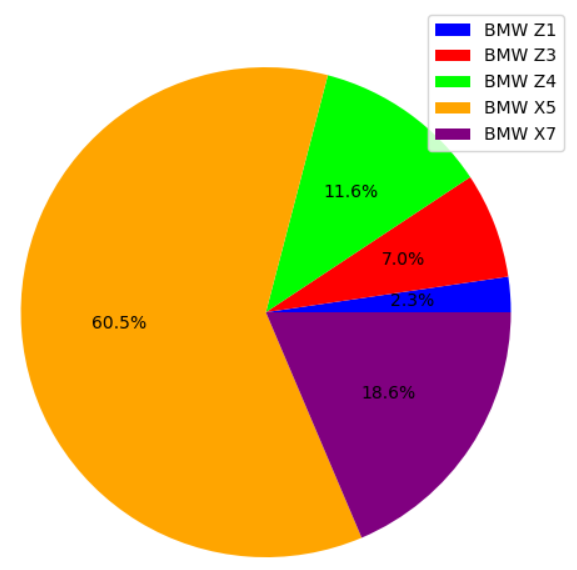
Разумеется, система не даст зарегистрировать новый автомобиль, если пользователь не заполнил хотя бы одно поле:



Если пролистать страницу ещё ниже, можно увидеть гистограмму по годам, в которые были произведены автомобили.



Сразу под ней расположена круговая диаграмма, отображающая процентное соотношение марок автомобилей в таблице.

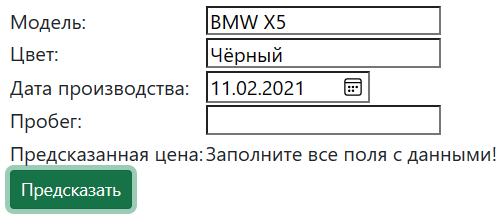


Под графиками расположено окно для предсказания цены автомобиля по заданным параметрам.

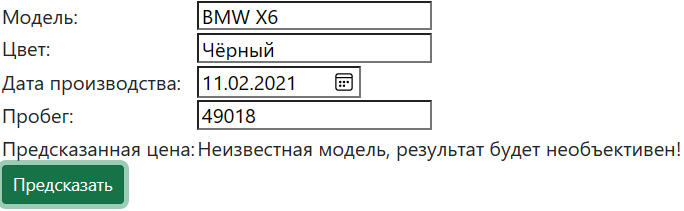
Изображение выглядит как текст, снимок экрана, линия, Шрифт

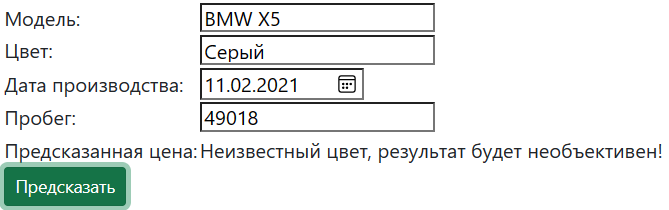
Автоматически созданное описание

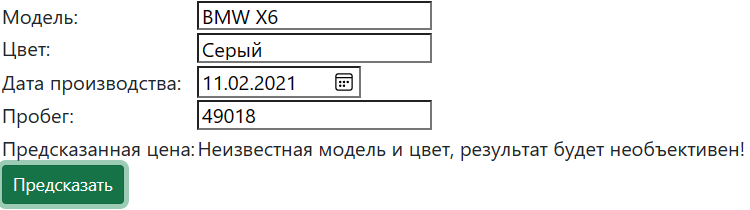
Необходимо заполнить все поля и нажать кнопку «Предсказать» для получения результата. Если заполнить не все поля, программа выведет соответствующее предупреждение.



Хочу ещё раз заострить внимание на том, что категориальные признаки «Модель» и «Цвет» имеют слишком большой весовой коэффициент. Если у обученной модели не будет данных хотя бы об одном из этих признаков, то программа выдаст соответствующее предупреждение.







В самом низу страницы расположена кнопка «Об авторе», которая при нажатии перекидывает на новую страницу.



