

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ»
(ГАПОУ СПО ОКЭИ)

КУРСОВОЙ ПРОЕКТ

ОКЭИ 09.02.07. 4323. №26 ПЗ
(код документа)

Разработка сайт аренды автомобилей

Количество листов 30
Дата готовности 27.12.23
Руководитель Гукасян А. Д.
Разработал Широнин Н.С.
Защищен _____ с оценкой _____
(дата)

Оренбург 2023

Содержание

Введение	3
1 Анализ предметной области	5
2 Проектирование приложения	8
3 Разработка программного обеспечения	10
3.1 Описание технологического стека разработки	10
3.2 Описание алгоритма работы	12
3.3 Описание интерфейса пользователя	13
4 Тестирование приложения	14
4.1 План тестирования	14
4.2 Оценка результатов проведения тестирования	16
Заключение	22
Список используемых источников	23
Приложение А (обязательное)	25
Приложение Б (обязательное)	26
Приложение В (обязательное)	27
Приложение Г (обязательное)	28
Приложение Д (обязательное)	30

					ОКЭИ 09.02.07. 4323. 26 0							
Изм.	Лист	№ докум.	Подпись	Дата								
Разраб.		Широнин Н.С.			Отчёт по практике по профилю специальности			Лит.	Лист	Листов		
Провер.		Гцкасян А. Д.								2	30	
Реценз								Отделение — очное гр. 4бд1				
Н. Контр.												
Утверд.												

Введение

Веб-приложение – это программное обеспечение, разработанное специально для работы в сети Интернет. Оно доступно пользователю через веб-браузер и предоставляет функциональность, которая включает в себя обработку данных, визуализацию информации, взаимодействие с базой данных, отправку и получение запросов и многое другое.

Разработка веб-приложения требует проведения нескольких этапов. В первую очередь необходимо определить его основную концепцию и функциональные требования. Затем следует разработать дизайн интерфейса и создать макеты страниц, чтобы визуализировать структуру приложения.

Они могут легко включать в себя инструменты для анализа данных и создания отчетов, что помогает в принятии информированных решений.

Пользователи могут использовать веб-приложения на разных устройствах, будь то компьютеры, смартфоны, планшеты или даже смарт-телевизоры, так как они не зависят от операционной системы.

Таким образом, веб-приложения широко используются в бизнесе, образовании, здравоохранении, государственном управлении, развлечениях и многих других сферах для улучшения производительности, доступности к информации и взаимодействия с пользователями.

В ходе курсовой должно быть решено несколько ключевых задач:

- разработка серверной части веб-приложения, включая создание интерфейса и определение функциональности;
- разработка системы.

Целью создания такого веб-приложения является улучшение доступности и удобства обслуживания клиентов.

Для успешной реализации намеченных задач требуется провести комплексную работу, которая включает в себя несколько ключевых этапов:

Прежде всего, необходимо определить и разработать логическую структуру будущего веб-приложения. Это включает в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, каким образом приложение будет собирать, хранить и обрабатывать.

Для успешной реализации намеченных задач требуется провести комплексную работу, которая включает в себя несколько ключевых этапов:

- определить и разработать логическую структуру будущего веб-приложения;

Это включает в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, как пользователи будут взаимодействовать с системой.

- для удобства использования веб-приложения необходимо разработать дизайн пользовательского интерфейса;

Это включает в себя выбор цветовой палитры, шрифтов, структуры страниц, расположения элементов управления и визуального оформления. Для

реализации данного пункта должны быть разработаны эскизы и макеты будущего веб-приложения.

– после разработки дизайна необходимо перейти к вёрстке страниц приложения;

Этот этап включает в себя создание Vue.js, Bootstrap кода для каждой страницы, определённой в логике приложения. Важно обеспечить адаптивность и кросс-браузерность, чтобы приложение корректно отображалось на различных устройствах и в разных браузерах.

– разработать архитектуру базы данных для разрабатываемой серверной части, включая выбор подходящих таблиц;

– завершив этап разработки архитектуры базы данных, перейдем к созданию серверной части.

Этот этап включает в себя подключение библиотеки ReadBeanPHP (ORM).

Для успешного решения этой проблемы необходимо тщательно спроектировать архитектуру приложения, уделяя особое внимание взаимосвязям между его компонентами, эффективности обработки данных и запросов пользователей, а также обеспечить масштабируемость при увеличении нагрузки.

Этот комплексный подход позволит создать полноценное приложение, которое удовлетворит потребности пользователей и поможет в достижении поставленных целей.

После завершения программирования необходимо протестировать приложение для выявления и исправления возможных ошибок и недочетов. Затем следует его развертывание на сервере, чтобы сделать его доступным для пользователей.

Разработка веб-приложений может быть долгим и сложным процессом, требующим совместной работы разработчиков, дизайнеров и тестировщиков. Однако, современные инструменты и технологии делают этот процесс более эффективным и удобным.

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

1 Анализ предметной области

Сайт аренды автомобилей (или каршеринг) – это платформа, предоставляющая услуги по краткосрочной аренде автомобилей. Это альтернатива личному автомобилю, общественному транспорту и такси.

Основные функции сайта аренды авто:

Регистрация и авторизация пользователей: предоставление возможности пользователям зарегистрироваться и авторизоваться на сайте, используя различные методы идентификации (например, номер телефона, email, аккаунт в социальных сетях).

Выбор автомобиля:

предоставление пользователю возможности выбрать автомобиль для аренды, указав его характеристики (класс, марка, модель, вместимость и т.д.).

Бронирование и аренда автомобиля:

пользователь должен иметь возможность забронировать и арендовать автомобиль в удобное для него время и место, с учетом доступных автомобилей и свободных слотов.

Оплата услуг:

предоставление различных способов оплаты аренды автомобиля (банковские карты, электронные кошельки, мобильные платежи и т.п.).

Управление аккаунтом:

возможность для пользователя управлять своим аккаунтом на сайте (редактировать личные данные, изменять пароль и т. д.).

Обратная связь и поддержка:

возможность для пользователей оставить отзыв или задать вопрос, а также получить быструю помощь от службы поддержки сайта.

Наши преимущества:

Широкий выбор автомобилей различных классов и брендов.

Гибкие условия аренды - от краткосрочной до долгосрочной.

Прозрачные и понятные тарифы без скрытых платежей и комиссий.

Удобное и быстрое бронирование онлайн с возможностью отмены или изменения даты аренды.

Персональный подход к каждому клиенту - мы всегда готовы помочь в выборе автомобиля и ответить на все Ваши вопросы.

Все автомобили проходят регулярное техническое обслуживание и находятся в отличном состоянии.

Наличие страховки и франшизы на случай повреждения или угона автомобиля.

Удобное расположение пунктов выдачи и возврата автомобилей в разных городах и странах.

Чтобы забронировать автомобиль, выберите подходящий автомобиль на нашем сайте и заполните форму бронирования. Укажите свои контактные данные, срок аренды и дополнительные услуги, если необходимо.

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

2 Проектирование приложения

Проектирование сайта аренды автомобилей включает в себя несколько основных этапов:

- анализ требований и определение целей (на этом этапе необходимо провести анализ рынка, изучить потребности целевой аудитории, определить основные цели и задачи сайта аренды автомобилей);

- проектирование структуры сайта (следующим шагом является создание структуры сайта, которая определяет навигацию, категории товаров, страницы продуктов и другие разделы);

- разработка дизайна (на этом этапе создается дизайн интерфейса сайта с учетом бренд-идентичности, эстетических предпочтений аудитории и удобства использования. Важно учесть адаптивность дизайна для удобного просмотра на различных устройствах);

- разработка функционала (затем необходимо определить функциональные возможности сайта аренды автомобилей, такие как поиск и фильтрация товаров, корзина покупок, оформление заказа, управление аккаунтом и др. Важно убедиться, что все функции работают правильно и без ошибок);

- разработка базы данных (для хранения информации о товарах, клиентах, заказах и других данных требуется создать базу данных и разработать соответствующую структуру);

- тестирование и отладка (перед запуском сайта необходимо провести тестирование всех функций и проверить их работоспособность. Ошибки и неполадки должны быть исправлены, чтобы обеспечить безупречную работу сайта аренды автомобилей);

- запуск и поддержка (после успешного завершения всех предыдущих этапов интернет-магазин готов к запуску. Однако поддержка и обновление магазина должны осуществляться на постоянной основе, чтобы удерживать его в актуальном состоянии и обеспечивать удобство пользования для клиентов).

Функционал сайта аренды автомобилей может включать в себя следующие возможности:

- поиск и фильтрация товаров (пользователь может искать и отфильтровывать товары по разным критериям, таким как цена, бренд, тип товара и т.д.);

- корзина покупок (пользователь может добавлять товары в корзину и управлять содержимым, включая изменение количества, удаление товаров и расчет общей стоимости);

- оформление заказа (пользователь может выбирать тип доставки, указывать адрес доставки и оплачивать товары);

- отзывы и рейтинги товаров (пользователь может оставлять отзывы о приобретенных товарах и оценивать их качество, что помогает другим покупателям принять решение);

– интеграция с платежными системами (интернет-магазин может поддерживать различные способы оплаты, такие как кредитные карты, электронные платежи и другие);

– подборки и рекомендации (на основе предпочтений и истории покупок пользователя магазин может предлагать персонализированные подборки товаров и рекомендации);

– программа лояльности (интернет-магазин может предлагать бонусы, скидки и специальные предложения своим постоянным покупателям).

Функции пользователя:

- просмотр товара;
- добавления товара в корзину;
- удаление товара;
- фильтры товара;
- заказ товара.

Функции Администратора:

- просмотр статистики;
- просмотр заказов;
- добавления товара;
- удаления товара;
- управления учётными записями пользователей.

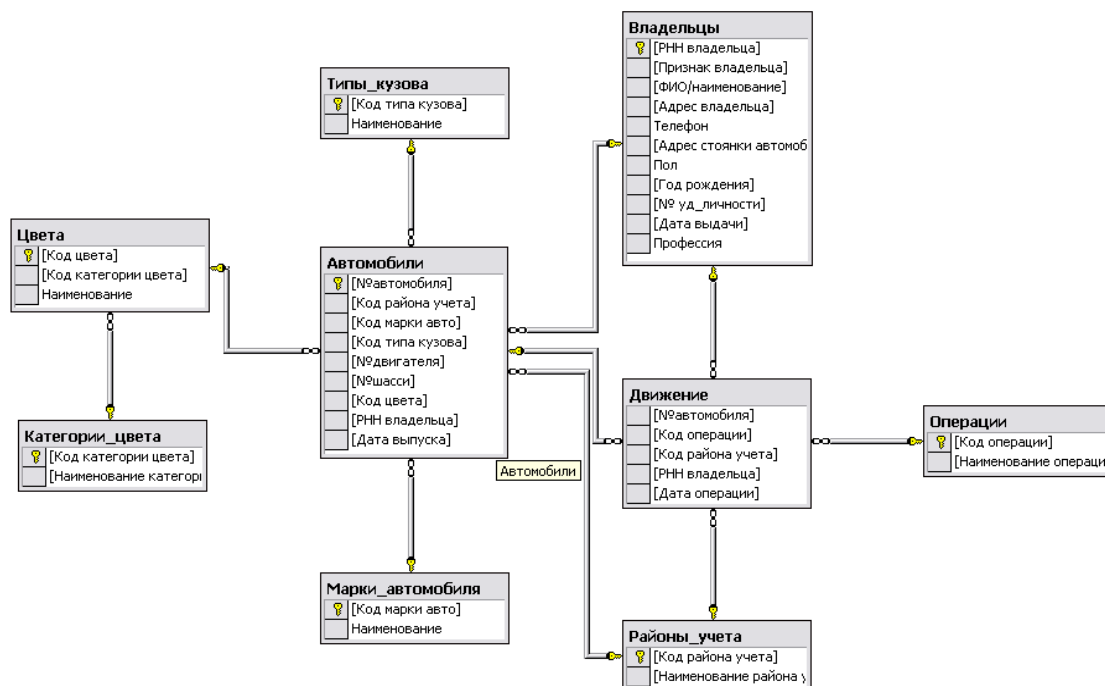


Рисунок 1 – Схема базы данных

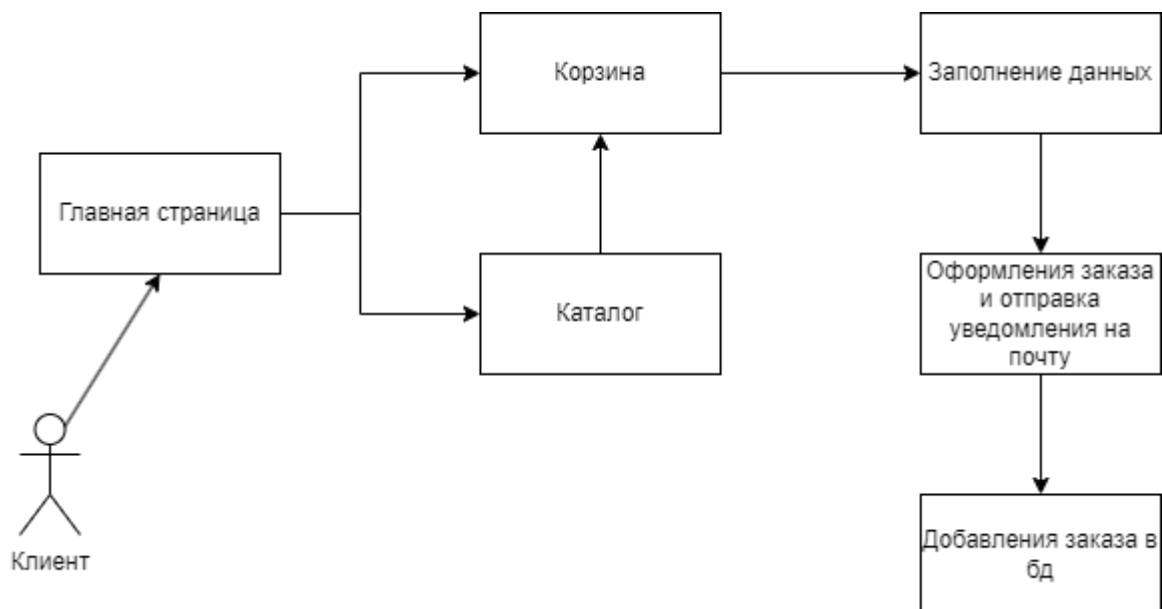


Рисунок 2 – Схема работы интернет магазина

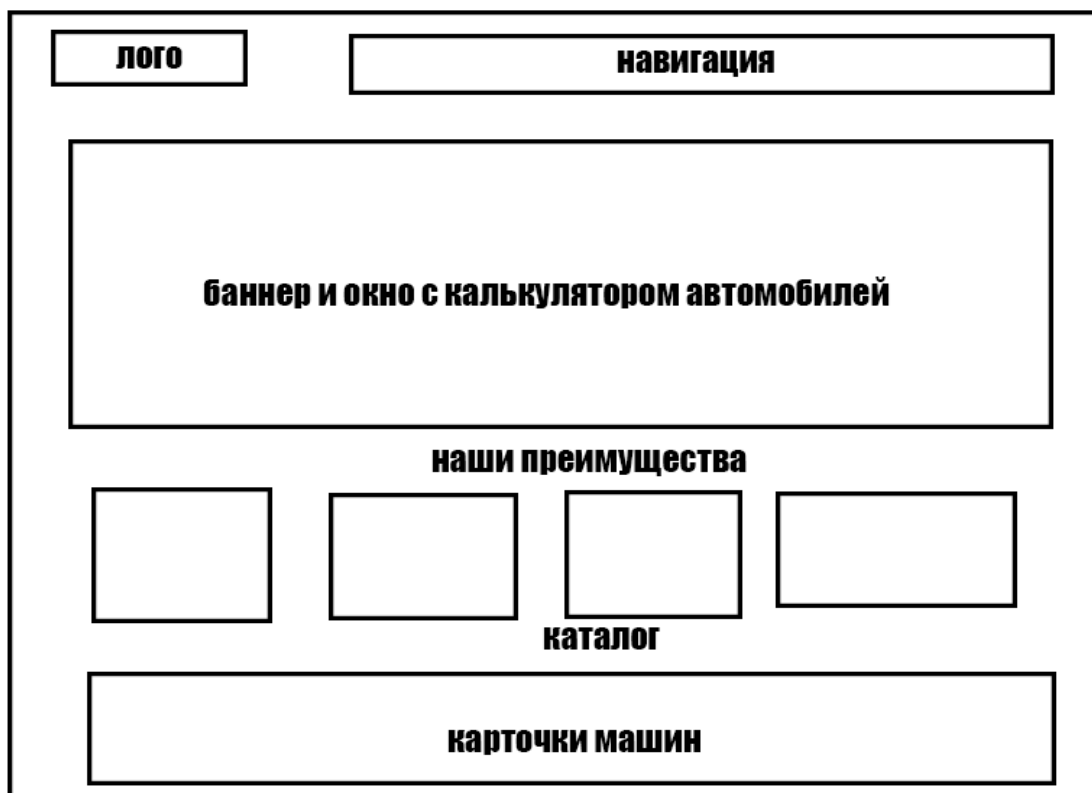


Рисунок 3 –Эскиз сайта аренды автомобилей

3 Разработка программного обеспечения

3.1 Описание технологического стека разработки

3.1.1 Среда разработки

В качестве интегрированной среды разработки рекомендуется использовать Visual Studio Code или PhpStorm. Для удобства работы с IDE могут использоваться такие плагины как Emmet.

3.1.2 Удаленный доступ

Для удаленного управления и развертывания приложения рекомендуется использовать SSH.

3.1.3 Технологии

Для реализации веб-приложения рекомендуется использовать веб-сервер Nginx или Apache.

В качестве языков программирования должны быть использованы:

PHP 8.0 для разработки серверной логики;

HTML для написания структуры сайта;

JavaScript, Vue.js для клиентской логики.

CSS для создания и изменения стиля элементов веб-страниц и пользовательских интерфейсов.

Для реализации некоторых функций могут быть использованы библиотеки и фреймворки:

PDO для упрощения работы с БД;

Bootstrap для быстрого написания CSS кода.

3.2 Описание алгоритма работы

Алгоритм работы интернет-магазина может выглядеть следующим образом:

- пользователь заходит на сайт интернет-магазина;
- пользователь может воспользоваться функцией поиска для быстрого нахождения нужного товара. Также можно использовать фильтры для уточнения результатов поиска;
- пользователь выбирает интересующий его товар и переходит на страницу с подробной информацией о товаре. Здесь пользователь может ознакомиться с описанием, характеристиками, ценой, наличием и отзывами о товаре;
- если товар удовлетворяет требованиям покупателя, он может добавить его в корзину;
- в корзине пользователь может изменять количество товаров, удалять ненужные товары или продолжить покупки, добавив еще товары в корзину;
- когда пользователь закончил выбор товаров, он переходит к оформлению заказа. Здесь ему предлагается заполнить форму с контактной информацией и выбрать способ доставки и оплаты.

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		9

- после заполнения формы данных и выбора всех деталей заказа, пользователь подтверждает заказ;
- после подтверждения заказа пользователю предоставляется информация о деталях заказа, и он может произвести оплату;
- когда оплата прошла успешно, интернет-магазин отправляет уведомление о заказе на электронную почту покупателя и начинает обработку заказа;
- заказ обрабатывается сотрудниками магазина. Товары подготавливаются к отправке, упаковываются и передаются службе доставки;
- покупатель получает заказ по выбранному им способу доставки;
- после получения товаров пользователь может оставить отзыв о покупке и оценить качество обслуживания.

3.3 Описание интерфейса пользователя

Интерфейс пользователя для сайта аренды автомобилей должен быть удобным, интуитивно понятным и привлекательным для пользователей. Он должен содержать следующие элементы:

Шапка сайта с логотипом компании, меню навигации по страницам (Главная, Каталог автомобилей, Услуги, Контакты и т.д.) и возможностью входа/регистрации пользователя.

Поиск автомобилей с возможностью фильтрации по различным критериям, таким как марка, модель, год выпуска, тип автомобиля, стоимость аренды и другие параметры.

Страница с каталогом автомобилей, где пользователь может просматривать доступные для аренды машины с описанием и фотографиями. Форма для онлайн бронирования автомобиля, где пользователь может выбрать даты аренды, дополнительные услуги (например, GPS навигация, детский креслец и др.) и оставить свои контактные данные.

Страница с информацией об услугах компании, ценах на аренду, правилах и условиях аренды, способах оплаты и другой полезной информацией.

Контактная информация компании, форма обратной связи, а также ссылки на социальные сети.

Подвал сайта с дополнительными ссылками на основные разделы сайта, информацией о компании и копирайтом.

Интерфейс пользователя должен быть адаптивным, чтобы сайт корректно отображался на всех устройствах (компьютерах, планшетах, смартфонах) и иметь приятный дизайн, который будет способствовать удобству пользования сайтом. Важно также обеспечить быструю загрузку страниц сайта и удобную навигацию для пользователей.

					ОКЭИ 09.02.07. 4323. 26 0	Лист 10
Изм.	Лист	№ докум.	Подпись	Дата		

4 Тестирование приложения

4.1 План тестирования

Для тестирования веб-приложения был выбрано несколько методов тестирования.

Предварительное тестирование:

- проверка совместимости интерфейса с различными браузерами (Chrome, Firefox, Safari, Edge и т. д.);
- проверка отзывчивого дизайна и адаптивности интерфейса на различных устройствах (компьютеры, планшеты, мобильные телефоны);
- проверка работы интерфейса на разных операционных системах (Windows, macOS, Android, iOS).

Тестирование функциональности:

- поиск и фильтрация товаров;
- просмотр детальной информации о товаре и добавление его в корзину;
- расчет общей стоимости заказа, применение скидок и промо-кодов;
- оформление заказа и оплата товаров;
- управление профилем пользователя, изменение персональных данных и адресов доставки;
- добавление и удаление товаров из списка желаемых.

Тестирование корзины и оформления заказа:

- проверка добавления товаров в корзину и изменения их количества;
- проверка удаления товаров из корзины;
- проверка вычисления общей стоимости заказа и применения доставки и налогов;
- проверка выбора и просмотра доступных способов доставки и оплаты;
- проверка правильной обработки данных пользователя (адрес доставки, контактная информация).

Тестирование процесса оформления заказа:

- проверка заполнения обязательных полей при оформлении заказа;
- проверка правильности работы расчета промежуточных и итоговых сумм заказа;
- проверка отправки подтверждения заказа на почту пользователя;
- проверка правильности работы системы платежей и получения платежных уведомлений.

Тестирование безопасности:

- проверка защиты информации пользователей (SSL-сертификат, шифрование данных);
- проверка безопасности платежных операций и передачи финансовых данных;
- проверка защиты от несанкционированного доступа к личным данным

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		11

пользователей.

Тестирование производительности:

- проверка скорости загрузки страниц, особенно на мобильных устройствах;
- проверка работы интерфейса при большом количестве одновременных пользователей;
- проверка работы при нагрузке на сервер и контрольные точки сети.

Тестирование на различных стадиях разработки и обновления:

- проверка корректности работы новых функций и изменений;
- проверка обратной совместимости с предыдущими версиями сайта;
- проверка наличия и работоспособности всех ссылок и переходов на другие страницы.

Тестирование системы обратной связи:

- проверка работоспособности формы обратной связи;
- проверка получения уведомлений о входящих сообщениях;
- проверка корректного отображения сообщений в административной панели.

Тестирование на разных языках:

- проверка локализации интерфейса на различные языки, включая русский, английский и т.д.;
- проверка корректного отображения и ввода символов для каждого языка.

Тестирование на мобильных устройствах:

- проверка отзывчивого дизайна и адаптивности интерфейса на различных мобильных устройствах;
- проверка корректности работы интерфейса на разных операционных системах (Android, iOS).

DDOS attack. Чат-бот может быть эффективным только в том случае, если он может быть устойчивым к DDOS атакам. Тестирование реализовано на рисунке 3.

Код программы тестирования DDOS атаки на веб-приложения представлен в приложении Д.

4.2 Оценка результатов проведения тестирования

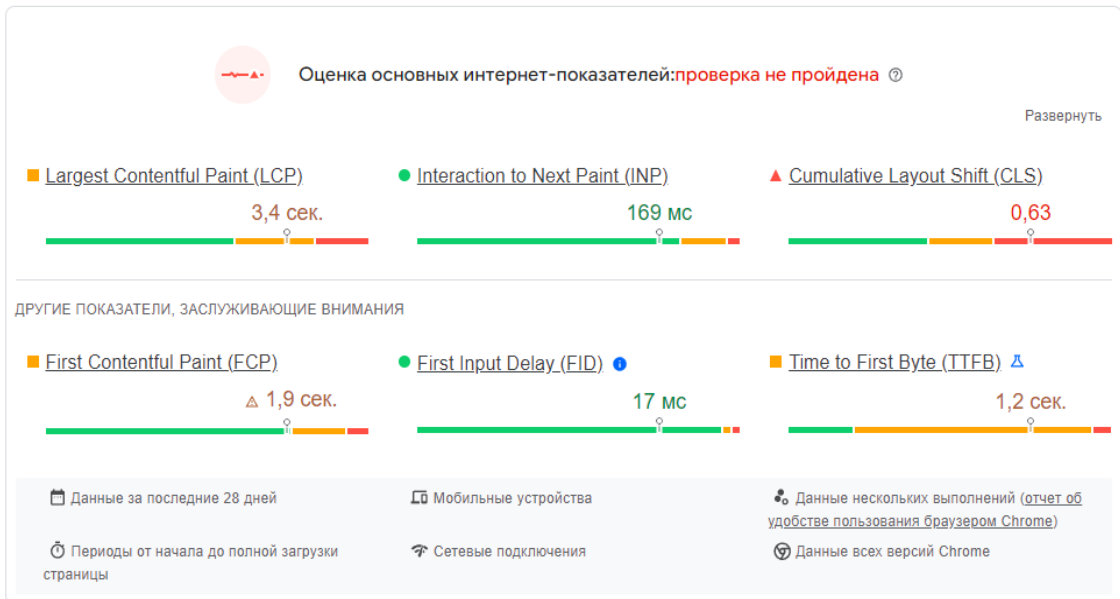


Рисунок 4 – тестирования ПК версии сайта

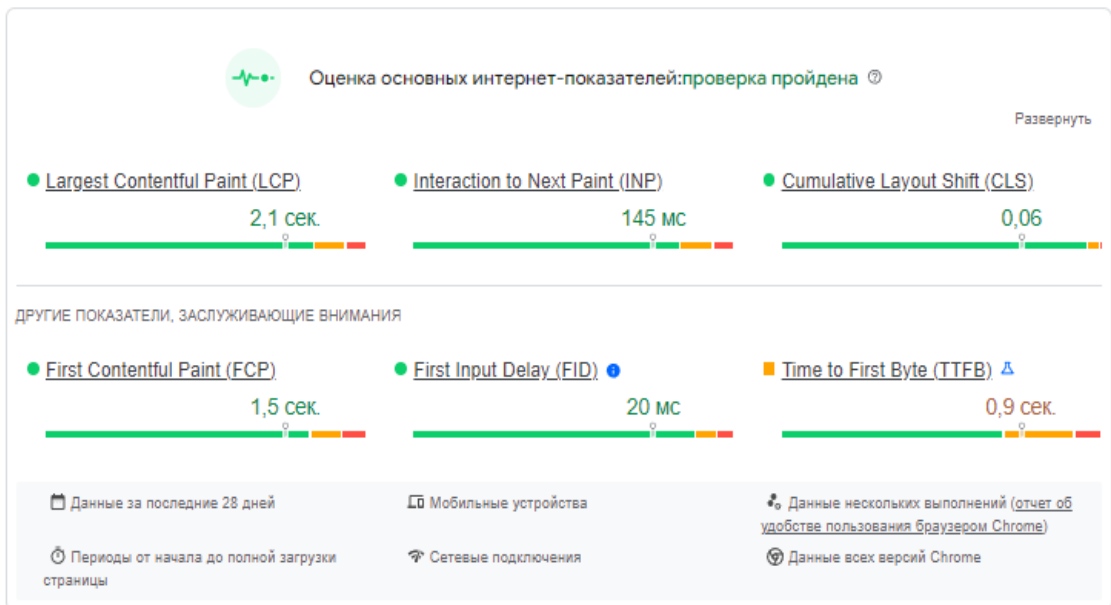


Рисунок 5 – тестирования мобильной версии сайта

Заключение

В ходе курсовой работы были выполнены работы по проектированию и разработке веб-приложения.

Для реализации веб-приложения был проведен анализ предметной области, составлено техническое задание, в котором определялись аппаратные и технические требования к разработке, была выполнена разработка проектной документации, в том числе подготовлены макеты и эскизы веб-приложения, реализован подбор графических материалов, после чего была выполнена разработка веб-приложения, включающая в себя реализацию шаблонов и динамических элементов веб-приложения и выполнен контрольный пример и были реализованы следующие функции:

- просмотр товаров;
- добавления товара;
- удаление товара;
- добавления, редактирование товара администратором;
- просмотр статистики;
- автоматизировать обработку запросов;
- осуществлять поиск и сбор данных.

В ходе курсовой работы были решены следующие задачи:

- определена и разработана логическая структура будущего веб-приложения, что включало в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, как пользователи будут взаимодействовать с системой;

- для удобства использования веб-приложения необходимо разработать дизайн пользовательского интерфейса. Это включает в себя выбор цветовой палитры, шрифтов, структуры страниц, расположения элементов управления и визуального оформления. Для реализации данного пункта были разработаны эскизы и макеты будущего веб-приложения;

- после разработки дизайна было решено перейти к верстке страниц приложения, что включало в себя создание Vue.js, Bootstrap кода для каждой страницы, определенной в логике приложения. Важно было обеспечить адаптивность и кросс-браузерность, чтобы приложение корректно отображалось на различных устройствах и в разных браузерах, что и было достигнуто;

- была произведена разработка архитектуры базы данных для разрабатываемой серверной части, включая выбор подходящих таблиц;

- завершив этап разработки архитектуры базы данных, был совершён переход к созданию серверной части. Этот этап включает в себя подключение библиотеки ReadBeanPHP (ORM) и написания API для веб-приложения.

Подводя итоги выпускной квалификационной работы, можно сделать вывод, что поставленная цель исследования была достигнута, качество обработки клиентских обращений стало выше, путём разработки и внедрения на хостинг в декабре 2024 года.

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		14

Список использованных источников

- 1 Алексеев Ю.М. Быстро и легко создаем, программируем, шлифуем и раскручиваем web-сайт: учеб. пособие / Ю.М. Алексеев. - М.: Лучшие книги, 2011. - 189 с.
- 2 Ахромов Я.В. Технологии Web-дизайна и Flash-технологии: учеб. пособие / Я.В. Ахромов. - Ростов н/Д: Феникс, 2009. - 265 с.
- 3 Басыров Р. 1С-Битрикс: постройте профессиональный сайт сами / Р. Басыров. - СПб.: Питер, 2010. - 237 с.
- 4 Белунцов В. Новейший самоучитель по созданию Web-страниц / В. Белунцов. - М.: NT Press, 2012. - 185 с.
- 5 Берд Дж. Веб-дизайн. Руководство разработчика: учебник. - СПб.: Питер, 2012. - 224 с.
- 6 Благодатских В.А. Стандартизация разработки программных средств: учеб. пособие / В.А. Благодатских. - М.: Финансы и статистика, 2011. - 544 с.
- 7 Бородаев Д.В. Web-сайт как объект графического дизайна: монография / Д.В. Бородаев. - Харьков: «Септима ЛТД», 2011. - 288 с.
- 8 Браун Брэдли Oracle Database. Создание Web-приложений: учебник / Брэдли Браун, Ричард Ниемик, Джозеф С. Треззо. - СПб.: Лори, 2011. - 722 с.
- 9 Смирнова И. Начала web-дизайна: учеб. пособие / И. Смирнова. - СПб.: БХВ-Петербург, 2009. - 231 с.
- 10 Советов Б.Я. Информационные технологии: учебник для ВУЗов / Б.Я. Советов, В.В. Цехановский. - М.: Высшая школа, 2010. - 263 с.
- 11 Стогний А.А. Основы компьютерной грамотности: учебник / А.А. Стогний. - Киев: Головное издательство, 2009. - 215 с.
- 12 Сырых Ю.А. Современный веб-дизайн. Рисуем сайт, который продает: учеб. пособие / Ю.А. Сырых. - М.: Вильямс, 2009. - 304 с.
- 13 Тихонов А.И. Публикация данных в Internet: учебник / А.И. Тихонов. - М.: Изд-во МЭИ, 2011. - 96 с.
- 14 Трофимов В.В. Информационные системы и технологии в экономике и управлении: учебник / В.В. Трофимов. - М.: Высшее образование, 2010. - 480 с.
- 15 Угринович Н.Д. Практикум по информатике и информационным технологиям / Н.Д. Угринович. - М.: Бином. Лаборатория занятий, 2012. - 400 с.
- 16 Программная среда для веб-разработки. OpenServer.: [Электронный ресурс], 2022 года – Режим доступа: <https://ospanel.io/> – 05.06.2022.
- 17 Денвер - локальный сервер. DENVER.: [Электронный ресурс], 2022 года – Режим доступа: <http://www.denwer.ru/> – 01.06.2022.
- 18 Когда display:none, а когда visibility: hiddenhtmlbook.ru.: [Электронный ресурс], 2022 года – Режим доступа: <http://htmlbook.ru/> – 25.04.2022.
- 19 HTML и Web дизайн для начинающих: [Электронный ресурс], 2022 года – Режим доступа: <http://sovet.h1.ru> – 29.04.2022.
- 20 Создание шаблона сайта с помощью CSS: [Электронный ресурс], 2022 года – Режим доступа: <http://ruseller.com> – 21.04.2022.

21 Создание страниц сайта с помощью HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://roumik.ru> – 21.04.2022.

22 Магия творчества. История вышивания: [Электронный ресурс], 2022 года – Режим доступа: <http://myhobby.net> – 21.04.2022.

23 Архитектурная мастерская. Создание сайта: [Электронный ресурс], 2022 года – Режим доступа: <http://wonderfulstyle.ucoz.com> – 21.04.2022.

24 Создание сайтов: HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://www.codeharmony.ru> – 21.04.2022.

25 Основы CSS и HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://www.web-lesson.ru> – 21.04.2022.

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		16

Приложение А

Информационная модель

(обязательное)

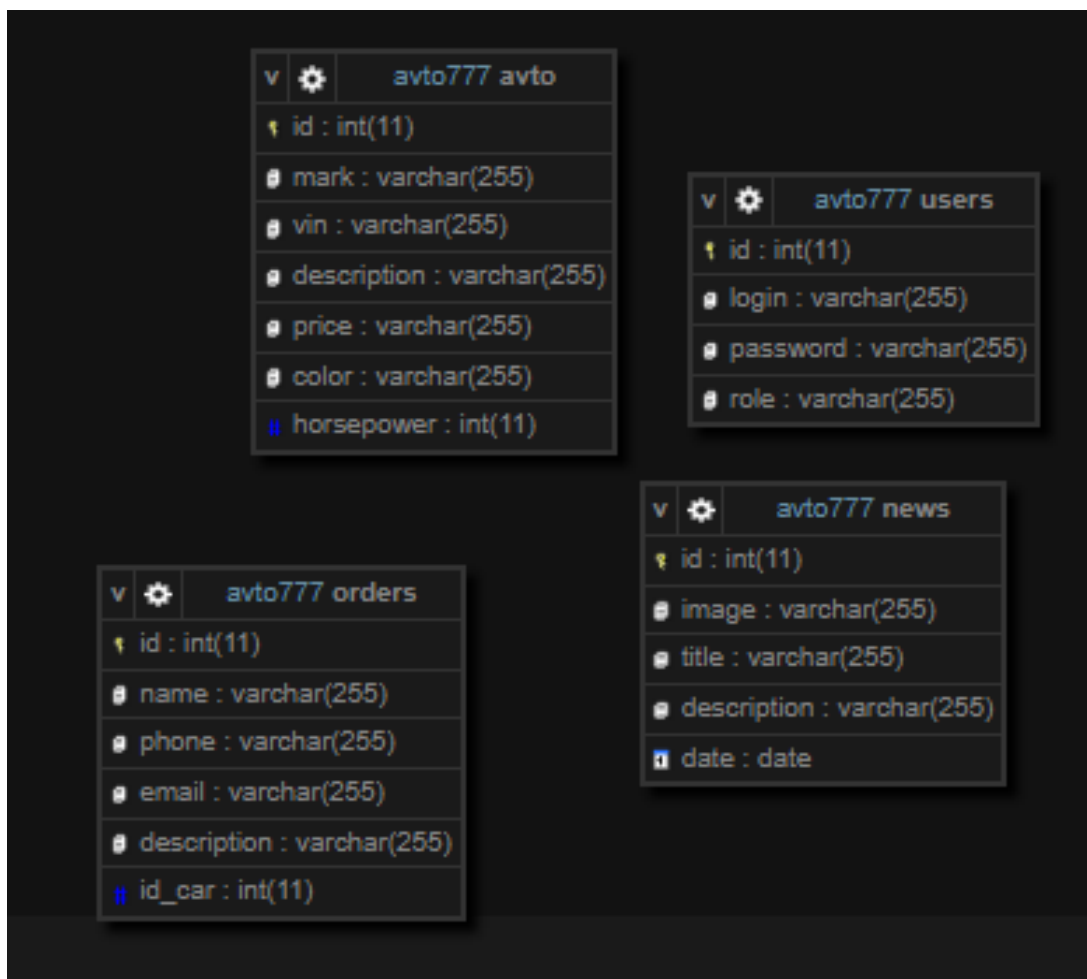


Рисунок А-1 – ER-диаграмма

Приложение Б

Функциональная модель (обязательное)

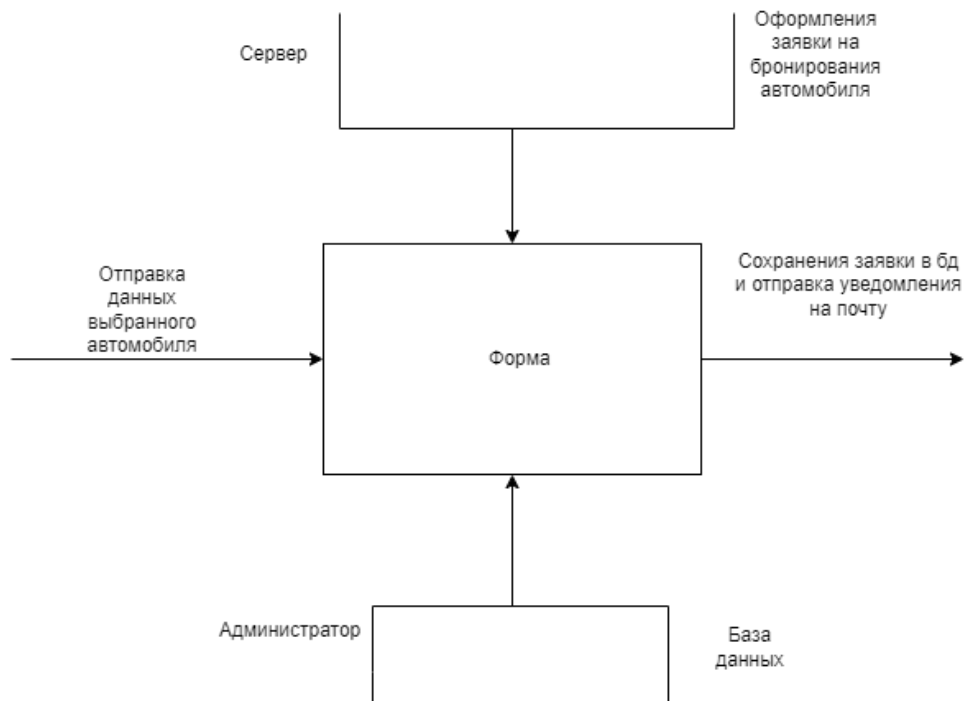


Рисунок Б-1 – Диаграмма контекстная диаграмма IDEF0

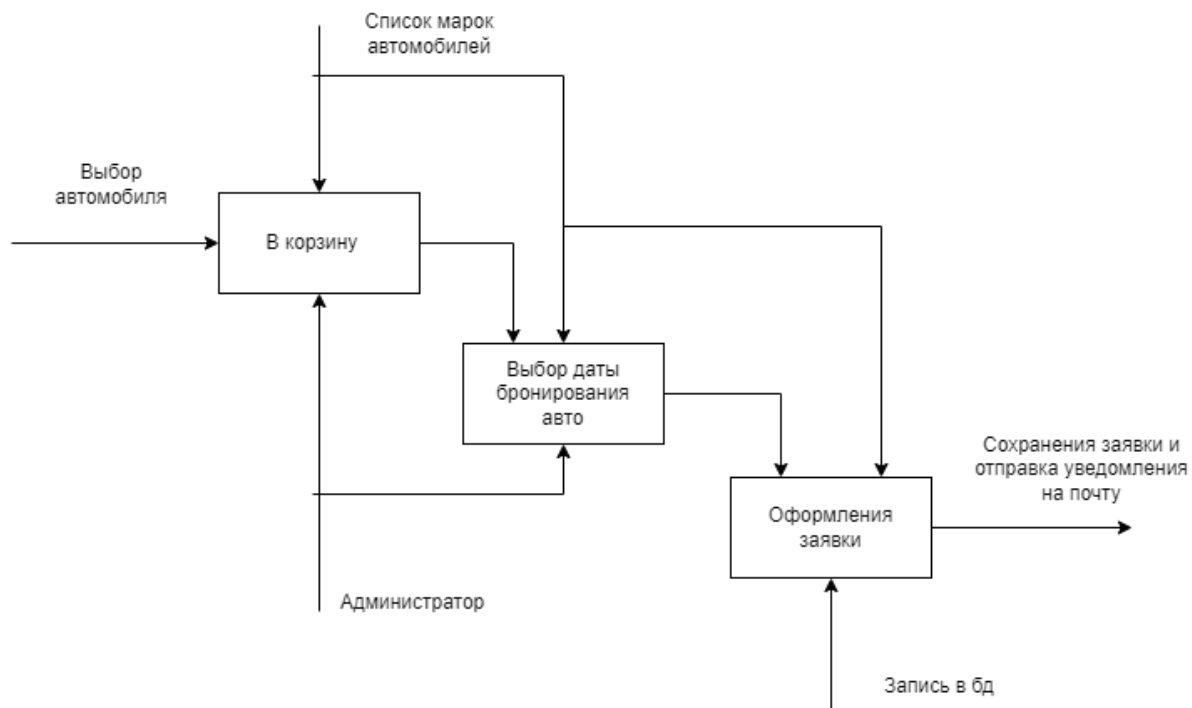


Рисунок Б-2 – Диаграмма декомпозиции первого уровня

Приложение В

Диаграмма прецедентов (обязательное)



Рисунок Б-3 – Диаграмма претендентов сайта аренды автомобилей

Приложение Г

Дизайн сайта

(обязательное)

Авто777

+7 (800) 333 08 33

Круглосуточно

Оформить заказ

Мой профиль

Прокат автомобилей
В Оренбурге

Подобрать автомобиль

Получения автомобиля

наш офис: г. Оренбург, Чкалова 11

▼

22.11.2022

📅

12:00

⌚

Возврат там же

наш офис: г. Оренбург, Чкалова 11

▼

22.11.2022

📅

12:00

⌚

Подобрать

Хотите арендовать автомобиль?

Арендуйте бесплатно

Коллекция шпал и ковшевых Спиритовых
шпал и ковшевых Спиритовых
Спиритовых. Предметы
предметы для всех городов
и стран, где есть шпалы ковшевых
шпалы. На шпалы ковшевых
шпалы ковшевых на шпалы 10% шпалы
шпалы ковшевых шпалы ковшевых, а шпалы
шпалы ковшевых 1 рубль.

ТОЛЬКО НОВАЯ

```

Все значения, которые принадлежат
к группе значений ValueOf Set
XenSet - не имеют. При
этом XenSet не является
aSetValueOf, так как не
является реализацией
стандарта value
значения на
элементыXenSetValueOf Set не
имеет.

```

Подача по городу

[illegible]

Гарантия низких цен

[illegible]

Рисунок Г-1 – Шаблон страницы главной страницы

					ОКЭИ 09.02.07. 4323. 26 0 -	Лист
Изм.	Лист	№ докум.	Подпись	Дата		20

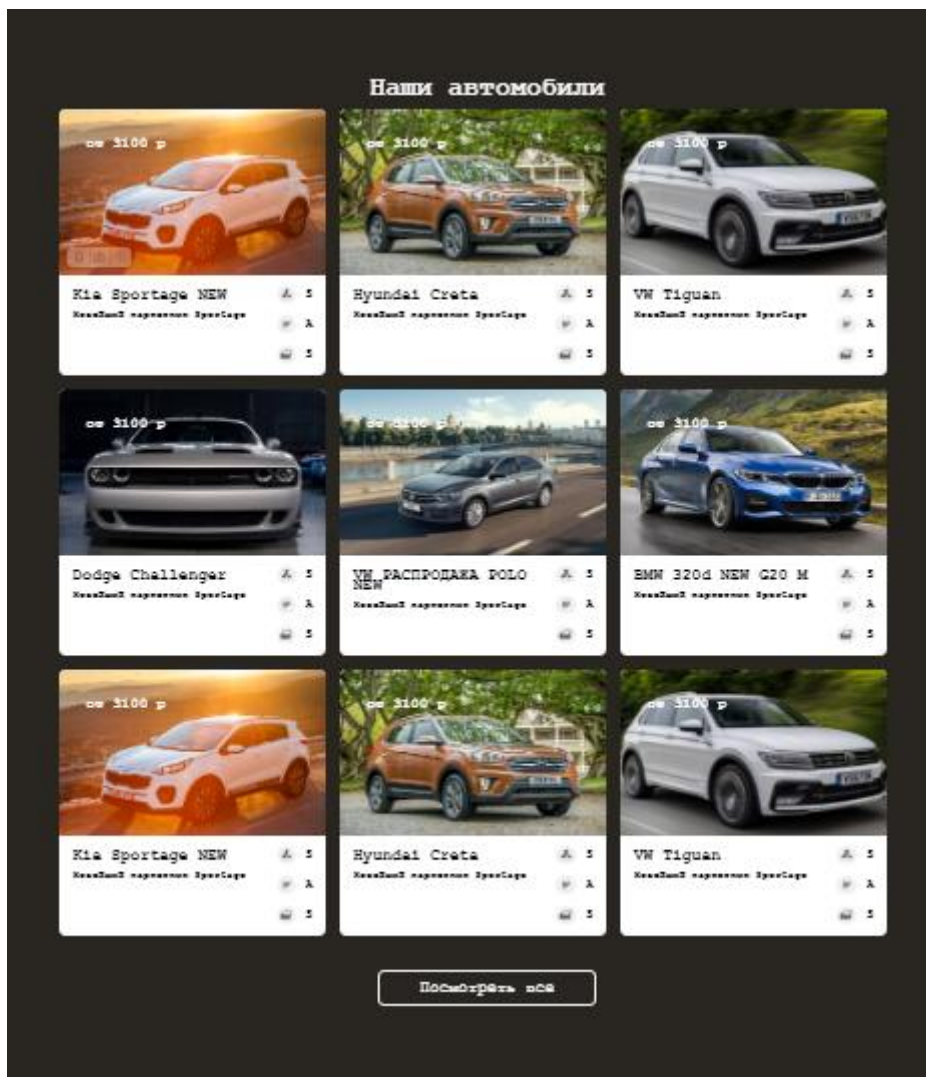


Рисунок Г-2 – Шаблон каталога

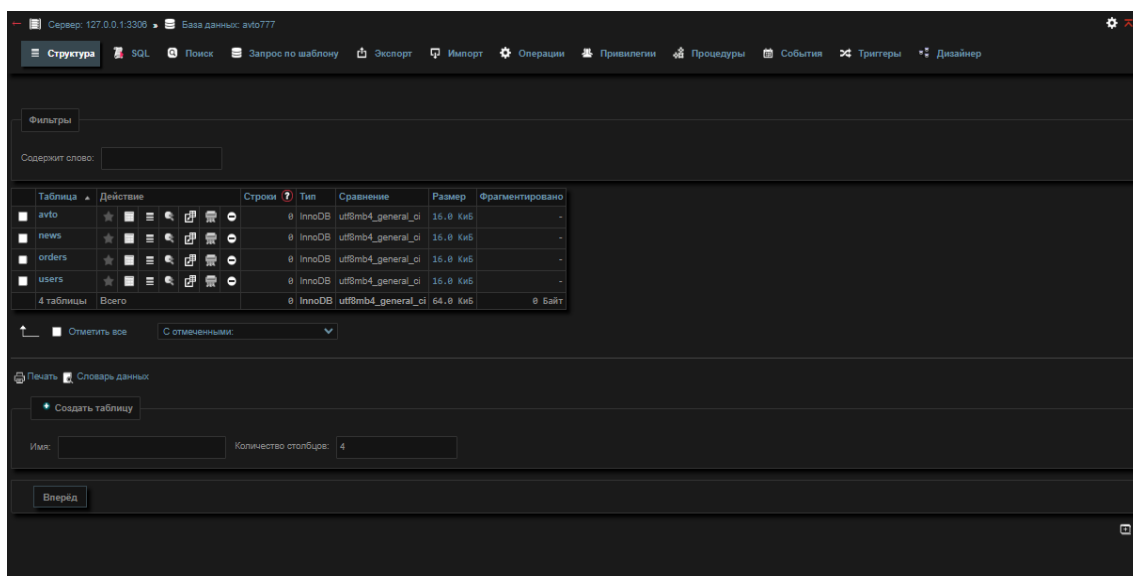


Рисунок Г-3 – Шаблон административной панели

Приложение Д

Листинг программы (обязательное)

```
<template>
  <div class="home">
    <div
      class="bg-home">
        <video
          autoplay=""
          loop="" muted=""
          style="filter: brightness(40%); margin: auto; position: absolute; z-
index: -1; top: 582px; left: 50%; transform: translate(-50%, -50%);
visibility: visible; opacity: 1; width: auto; height: 962px;">
          <source src="https://storletcar.com/video/main-bg.mp4"
type="video/mp4">
          <source src="https://storletcar.com/video/main-bg.webm"
type="video/webm">
          <source src="https://storletcar.com/video/main-bg.ogv"
type="video/ogg">
        </video>
        <vPanel />
      </div>

    <div class="v-container">
      <vItem />
      <div class="v-container__info">
        <div class="v-inner v-item__info">
          
          <div>
            <div
              class="v-inner"
              style="margin-bottom: 10px;">
              
              
            </div>
            
          </div>
        </div>
      </div>
    </div>
    <div class="v-my__car">
      <div class="v-container">
```

```

    <h2>Наши автомобили</h2>
    <div class="v-inner__my__car">
      <vCard
        v-for="i in items"
        :image="i.image"
        :title="i.title"
        :description="i.description"
        :price="i.price"
        :key="i" />
    </div>
    <router-link
      class="v-btn__link"
      to="">Посмотреть все</router-link>
    </div>
  </div>
</template>

<script>
import vItem from '@components/Item/v-item.vue'
import vCard from '@components/Card/v-card.vue';
import vPanel from '@components/Panel/v-panel.vue';

export default {
  components: {
    vItem,
    vCard,
    vPanel
  },

  data() {
    return {
      items: [

      ]
    }
  },
}
</script>

<style scoped>

.v-container__info {
  max-width: 1184px;
  margin: 100px auto;
}

```

```

.v-item__info img {
  border-radius: 10px;
}

.bg-home {
  height: 800px;
}

.v-my__car {
  background: #292521;
  padding: 100px 0px;
  color: #fff;
}

.v-my__car > div > h2 {
  font-size: 39px;
  text-align: center;
}

.v-inner__my__car {
  display: flex;
  flex-wrap: wrap;
  align-content: flex-start;
}

.v-btn__link {
  font-size: 24px;
  text-decoration: none;
  display: flex;
  justify-content: center;
  color: #fff;
  border: 1px solid #fff;
  max-width: 300px;
  margin: 40px auto;
  padding: 10px;
  border-radius: 10px;
}

</style>

```

```

<template>
  <div class="v-card">
    <div class="v-image">
      <img

```



```

        :src="image"
        width="400"
        height="250" alt="">
        <span class="v-card__price">от {{price}} р</span>
    </div>
    <div class="v-card__inner v-color">
        <div class="v-card__data">
            <h2 class="v-card__title">{{ title }}</h2>
            <p>{{ description }}</p>
        </div>
        <div class="v-card__icon">
            <div class="v-inner">
                
                <h3>5</h3>
            </div>
            <div class="v-inner">
                
                <h3>A</h3>
            </div>
            <div class="v-inner">
                
                <h3>5</h3>
            </div>
        </div>
    </div>
</template>

<script>

export default {
  props: {
    title: String,
    description: String,
    image: String,
    price: String
  }
}

</script>

<style scoped>
.v-card {

```

```

width: 400px;
height: 400px;
background: #fff;
margin: 10px;
border-radius: 10px;
position: relative;
}

.v-card__inner {
display: flex;
justify-content: space-between;
align-items: flex-start;
margin: 0px 20px;
}

.v-card__price {
position: absolute;
left: 40px;
top: 40px;
font-size: 22px;
font-weight: 700;
}

img {
border-radius: 10px 10px 0px 0px;
}

.v-color {
color: #000;
}

.v-card__icon > div {
width: 55px;
}
</style>

```

```

<template>
  <div class="v-header">
    <div class="v-container">
      <div class="v-inner">
        
        <div class="v-phone">
          <h2>+7 (800) 333 08 33</h2>
          <p>Круглосуточно</p>
        </div>
      </div>
    </div>
  </div>
</template>

```

```

        <div class="v-btn">
            <router-link class="v-btn__link" to="">Обратный
звонок</router-link>
            <router-link class="v-btn__link" to="">Мой
профиль</router-link>
        </div>
    </div>
</div>
</template>

<style scoped>
.v-header {
    background: #292521;
    padding: 20px 0px;
}

.v-phone {
    color: #fff;
}

.v-btn__Link {
    border: 2px solid #808080;
    color: #fff;
    font-size: 20px;
    padding: 12px 30px;
    margin: 0px 10px;
    text-decoration: none;
    border-radius: 30px;
}

</style>

```

```

<template>
    <div class="v-window">
        <div class="v-window__title">
            <h2>Прокат автомобилей</h2>
            <h2 style="color: yellow;">В Оренбурге</h2>
        </div>
        <div class="v-panel">
            <h2 class="v-panel__title">Подобрать автомобиль</h2>
            <div class="v-panel__form">
                <h3 class="v-form__title">Получения автомобиля</h3>
                <div class="v-inner v-form__params">
                    <select name="" id="">

```

```

        <option value="">Наш офис: г. Оренбург, Чкалова
11</option>

        <option value="">Удобный вам аэропорт</option>
        <option value="">Удобный вам вокзал</option>
        <option value="">Любой удобный вам адрес</option>
        <option value="">Любой удобный вам адрес</option>
    </select>
    <input type="date">
    <input type="time">
</div>
<h3 class="v-form__title">Возврат там же</h3>
<div class="v-inner v-form__params">
    <select name="" id="">
        <option value="">Наш офис: г. Оренбург, Чкалова
11</option>

        <option value="">Удобный вам аэропорт</option>
        <option value="">Удобный вам вокзал</option>
        <option value="">Любой удобный вам адрес</option>
        <option value="">Любой удобный вам адрес</option>
    </select>
    <input type="date">
    <input type="time">
</div>
    <button class="v-form__btn">Подобрать</button>
</div>
</div>
</div>
</template>

<style scoped>
.v-window__title {
    text-align: center;
    margin-top: 230px;
}

.v-window__title h2 {
    font-size: 60px;
    line-height: 50px;
    color: #fff;
}

.v-window__title > h2:last-child() {
    color: yellow;
}

.v-panel {

```

```

background: #fff;
max-width: 800px;
margin: 40px auto;
height: 400px;
border-radius: 18px;
}

.v-panel__title {
text-align: center;
padding-top: 30px;
padding-bottom: 20px;
font-size: 30px;
}

.v-panel__form {
margin: 30px 40px;
}

.v-form__title {
margin: 30px 0px;
font-size: 24px;
}

select, input {
border: 0px;
border-bottom: 1px solid #000;
margin: 0px 5px;
font-size: 19px;
outline: none;
cursor: pointer;
}

.v-form__btn {
padding: 15px 30px;
font-size: 22px;
border: 0px;
color: #000;
background: rgb(255, 255, 0);
border-radius: 20px;
cursor: pointer;
margin-top: 30px;
transition: .3s;
}

.v-form__btn:hover {
background: rgb(227, 230, 66);
}

```

```
}
</style>
```

```
<template>
  <div class="v-footer">
    <div class="v-container">
      <div class="v-footer__inner">
        <div class="v-footer__item">
          
          <p align="justify">Информация на данном сайте носит
исключительно ознакомительный характер и не является публичной офертой,
определяемой положениями Статьи 437 Гражданского кодекса РФ.</p>
          <p>(C) 2016 «Avto777 Group» – аренда автомобилей в
Оренбурге</p>
        </div>
        <div class="v-footer__item">
          <a href="#">Выбор автомобиля</a>
          <a href="#">Станции аренды</a>
          <a href="#">Дисконтная система</a>
          <a href="#">Франчайзинг</a>
          <a href="#">Оперативный лизинг</a>
          <a href="#">Аренда с выкупом</a>
          <a href="#">0 компании</a>
          <a href="#">Политика безопасности и оплаты</a>
        </div>
        <div class="v-footer__item">
          <h3>Контакты</h3>
          <p>г. Оренбург, Чкалова 11<br>
8 (800) 333 08 33<br>
car56@orenburg.com</p>
        </div>
        <div class="v-footer__item">
          <h2>Подписывайтесь на рассылку
и узнавайте первым об акциях
и спецпредложениях</h2>
          <button class="v-footer__btn">Подписаться</button>
        </div>
      </div>
    </div>
  </div>
</template>

<style scoped>
.v-footer {
```

```

background: #292521;
border-top: 3px solid #fff;
padding: 40px;
}

.v-footer__inner {
display: flex;
justify-content: space-between;
align-items: flex-start;
}

.v-footer__item {
display: grid;
width: 300px;
}

.v-footer__item p, .v-footer__item a {
color: #fff;
margin: 15px 0px;
}

.v-footer__item a, .v-footer__item h2, h3 {
color: #fff;
margin: 4px 0px;
}

.v-footer__item h3 {
font-size: 22px;
}

.v-footer__item h2 {
font-size: 20px;
}

.v-footer__btn {
padding: 15px 30px;
font-size: 22px;
border: 0px;
color: #fff;
border: 2px solid rgb(255, 255, 0);
border-radius: 20px;
cursor: pointer;
margin-top: 30px;
transition: .3s;
background: transparent;
}

```

```
.v-footer__btn:hover {  
  color: #000;  
  background: rgb(227, 230, 66);  
}  
</style>
```

					ОКЭИ 09.02.07. 4323. 26 0	Лист
Изм.	Лист	№ докум.	Подпись	Дата		32

