

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК
КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
ИНФОРМАЦИОННЫХ СИСТЕМ

РАЗРАБОТКА WEB-ПРЕДСТАВИТЕЛЬСТВА ДЛЯ ИП КОРЕНЕВА И.В.

Выпускная квалификационная работа
обучающейся по направлению подготовки 02.03.03 Математическое
обеспечение и администрирование информационных систем
очной формы обучения, группы 07001402
Кирияновой Евгении Васильевны

Научный руководитель
к.т.н., доцент
Муромцев В.В.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1. АНАЛИТИЧЕСКАЯ ЧАСТЬ	6
1.1 Анализ деятельности предприятия	6
1.2 Обоснование целесообразности разработки	9
1.3 Анализ требований к разработке.....	10
1.3.1 Общие требования к сайту.....	10
1.3.2 Требования к функциям сайта.....	11
1.3.3 Требования к программному и техническому обеспечению	11
1.3.4 Разработка дерева функций	12
1.3.5 Разработка сценария диалога.....	14
1.4 Постановка задачи	16
2. ПРОЕКТНАЯ ЧАСТЬ.....	18
2.1 Описание технологий создания сайтов.....	18
2.2 Обоснование выбора технологий создания сайта	21
2.3 Построение функциональной модели системы	26
2.4 Разработка структуры сайта.....	30
2.5 Разработка структуры базы данных	32
3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	38
3.1 Описание программных модулей.....	38
3.2 Дизайн веб-представительства	43
3.3 Тестирование	53
ЗАКЛЮЧЕНИЕ	59
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	62
ПРИЛОЖЕНИЕ	64

ВВЕДЕНИЕ

Сегодня глобальная сеть Интернет выступает в качестве бизнес-инфраструктуры представляющей собой многомиллионный рынок потенциальных покупателей и продавцов. В этой связи предприятия различной бизнес-направленности создают онлайн веб-представительства для участия в этом рынке в целях собственной коммерческой выгоды.

Веб-представительство в общем случае мы будем определять как совокупность веб-страниц, доступных по доменному имени, поддерживаемых для полноценного представления информации, необходимых электронных документов, ведения интерактивного взаимодействия со сторонними пользователями как частными лицами, так и организациями с целью обеспечения представительства собственных интересов в сети Интернет. В данном случае веб-представительство предполагает наличие Интернет-магазина.

Веб-страницы должны удовлетворять определенным критериям, а именно должны сопровождаться и принадлежать предприятию, использоваться для коммерческого взаимодействия предприятия со своими клиентами (распространение сведений о товарах и услугах предприятия, ведение электронной торговли и т.д.), кроме этого веб-страницы должны иметь свой URL-адрес и при этом быть связанными между собой гиперссылками.

Как было отмечено выше веб-представительство для ИП Коренева И.В. должно представлять собой информационный ресурс, с функционалом Интернет-магазина, на котором клиенты смогут быстро ознакомиться с перечнем автомобильных товаров, их характеристиками, стоимостью, получить возможность правильного подбора товаров для личного автомобиля, получить сведения об оформлении заказа, гарантиях, вариантах доставки, возврате и обмене товаров, осуществить оформление заказа и его

оплату, написать вопрос менеджеру, заказать обратный звонок, ознакомиться со схемой проезда к торговым точкам и т.д.

Веб-представительство – это оптимальная рекламная площадка, т.е. средство проведения рекламных кампаний и акций в сети Интернет.

Важным преимуществом веб-представительства является его круглосуточная работа, информация доступна покупателям автомобильных товаров 24 часа в сутки и семь дней в неделю.

Веб-представительство для ИП Коренева И.В. обеспечит положительный имидж предприятия, поскольку следование «в ногу» со временем и прогрессом – хороший тон в современном деловом мире рыночных отношений.

Объектом исследования работы является предприятие ИП Коренева И.В.

Предмет исследования – разработка веб-представительства для ИП Коренева И.В.

Цель работы – разработка веб-представительства (веб-магазина) для предприятия ИП Коренева И.В.

В соответствии с целью выделены следующие задачи:

1. Анализ предметной области.
2. Разработка проекта веб-представительства.
3. Реализация веб-представительства.
4. Тестирование разработанного программного обеспечения.

В ходе выполнения дипломной работы были использованы следующие методы исследования: анализ, обобщение, методы представления информации графическим способом, компьютерное моделирование и др. При этом использовалась специальная и техническая литература, справочные руководства, техническая документация, электронные источники информации, а также данные, полученные с помощью прикладного программного обеспечения и т.д.

Новизна и практическая значимость дипломного проекта состоит в практической разработке веб-представительства для ИП Коренева И.В.

1. АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Анализ деятельности предприятия

Веб-представительство разрабатывается для индивидуального предпринимателя Коренева И.В. в г. Белгород, т.е. физического лица, зарегистрированного в установленном законом порядке и осуществляющего предпринимательскую деятельность без образования юридического лица. Важной особенностью осуществления предпринимательской деятельности в качестве индивидуального предпринимателя является факт, что гражданин отвечает по обязательствам принадлежащим ему имуществом, за исключением имущества, на которое в соответствии с законом не может быть обращено взыскание.

Государственная регистрация и дальнейшая деятельность индивидуальных предпринимателей регламентируется Федеральным Законом РФ «О государственной регистрации юридических лиц и индивидуальных предпринимателей» N 129-ФЗ, Гражданским кодексом РФ, другими федеральными законами РФ, а также отдельными постановлениями Правительства РФ.

Предприятие ИП Коренева И.В. («Автошины у Ирины») начало свою деятельность в 2012 году. На сегодняшний день предприятие занимается поставкой и продажей автомобильных шин (летних; зимних: шипованных, не шипованных; всесезонных), дисков (литых, штампованных, кованных) из Японии, Кореи, Китая, России, Украины и других стран. ИП Коренева И.В. предлагает обширный ассортимент данных товаров от большинства ведущих мировых производителей. В наличии имеется более шести тысяч наименований товаров, ассортимент которых постоянно обновляется и пополняется новыми брендами и новыми моделями. Тщательный отбор

поставщиков позволяет вести продажу только лучших и современных изделий автомобильного колесного рынка.

Товар, предлагаемый клиентам, напрямую импортируется от производителя и храниться в складских помещениях предприятий, что позволяет сохранять конкурентно-способные цены на все товары. Предприятие сотрудничает не только с розничными, но и с оптовыми покупателями из разных регионов России. Для таких клиентов предлагаются специальные цены. С помощью транспортных компаний выполняется доставка в любой город России.

Предприятие имеет торговые магазины (склады) по двум адресам (ул. Корочанская, 150, ул. Калинина, 55), а также два «модуля» (две точки продаж) на авторынке (ул. Студенческая, 1д). Организационно-штатная структура предприятия представлена на рисунке 1.1.

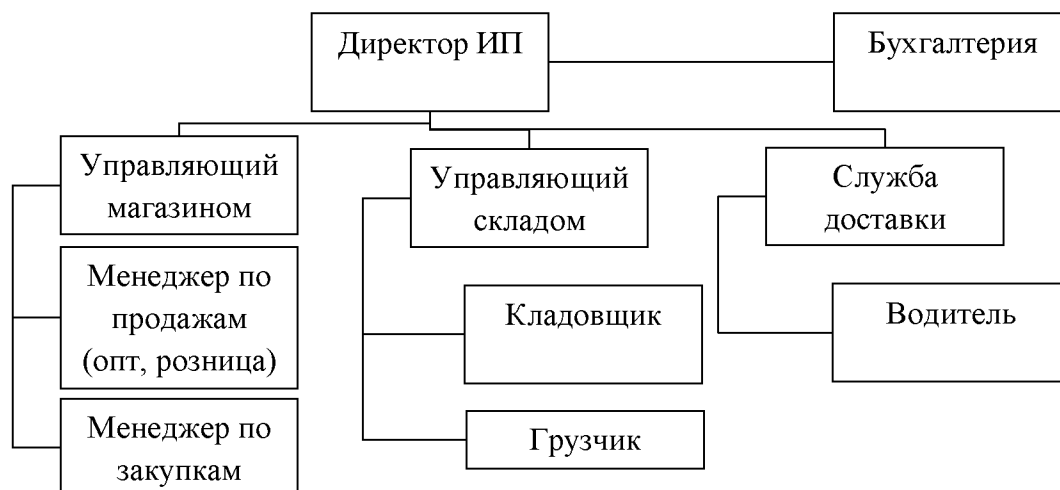


Рис. 1.1 Организационная структура ИП Коренева И.В.

Каждое подразделение в структуре решает определенный перечень задач. Директор осуществляет руководство текущей хозяйственно-финансовой деятельности предприятия и подразделений, обеспечивает эффективное взаимодействие структурных подразделений и т.д. В состав бухгалтерии входит: старший бухгалтер, бухгалтер. Основные функции бухгалтерии: планирование и организация бухгалтерского учета на предприятии, контроль над использованием ресурсов предприятия, контроль

над состоянием расчетов предприятия, а также сохранностью денежных средств, товарно-материальных ценностей и т.д.

Управляющие магазинов и складских помещений, в том числе служба доставки, занимаются изучением спроса и рекламы, обеспечивают реализации и сбыта продукции, услуг предприятия, обеспечивают выполнение заданий и обязательств по поставкам автомобильных товаров, выполняют сбор, обработку и анализ информации о деятельности российских и зарубежных организаций, работающих в сфере интересов предприятия, проводят необходимые мероприятия по заключению договоров с поставщиками товаров. Для обеспечения работы Интернет-магазина необходима новая должность – техник-программист. Сегодня наличие Интернет-магазина – жизненная необходимость для успешного ведения бизнеса, в частности расширения клиентской базы. В ходе анализа деятельности предприятия ИП Коренева И.В. выявлена потребность в разработке веб-представительства. Исходя из этого предложено новая организационная структура управления со специализированной IT-службой и должностью.

Схема новой организационной структуры будет иметь вид как на рисунке 1.2.

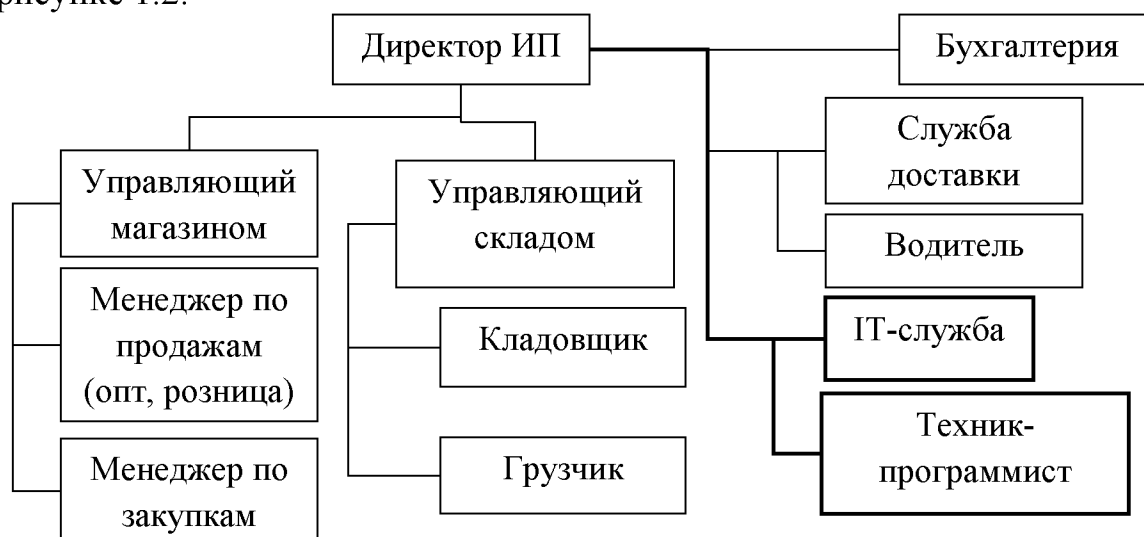


Рис. 1.2 Новая организационная структура ИП Коренева И.В.

К функциям техника-программиста должны относиться не только обязанности по обслуживанию сетевой инфраструктуры предприятия и аппаратно-программной базы, но и в частности разработка веб-представительства предприятия и его дальнейшее обслуживание и сопровождение.

Более детальное обоснование целесообразности разработки веб-представительства и интернет магазина представлено в следующем разделе.

1.2 Обоснование целесообразности разработки

Целесообразность разработки веб-сайта для ИП Коренева И.В. вызвана целью увеличения прибыли предприятия, а именно в связи с желанием увеличить клиентскую базу, более широко охватить ее территориально, заявить о себе как о предприятии в целом по России.

Другими словами, необходимо проведение продаж посредством сети Интернет, ведение рекламной деятельности онлайн, удаленное взаимодействие с клиентами.

Клиент в свою очередь имеет возможность сэкономить свое время на поиск и приобретение удовлетворяющего определенным характеристикам товаров, задать вопрос онлайн посредством не только стационарного компьютера, но и любого мобильного устройства, имеющего браузер и выход в Интернет.

Используя интерактивный каталог и функциональный поиск с набором фильтров на веб-сайте предприятия, клиент всегда сможет купить шины и диски самых разных производителей, размеров и функциональности.

Поскольку розничные продажи предприятие осуществляет через свои магазины, то с помощью веб-сайта ИП Коренева И.В. надеется на увеличение численности клиентов, на улучшение взаимодействия с ними, на лучшее

представление товаров для каждого человека независимо от его места жительства. Разработка веб-сайтов осуществляется, главным образом, с целью позиционирования, перепозиционирования (ребрендинг), повышения узнаваемости бренда, вывода на рынок нового продукта, мотивации пользователя к покупке. Как правило, каждый сайт одновременно решает несколько задач, но не более двух или трёх.

1.3 Анализ требований к разработке

1.3.1 Общие требования к сайту

Веб-представительство предприятия для ИП Коренева И.В. должно представлять собой информационный Интернет-магазин, на котором клиенты в общем случае будут иметь следующие возможности:

1. Получить удобную возможность совершения покупки выбранного автомобильного товара (оформление, оплата).
2. Быстро ознакомиться с перечнем автомобильных товаров, их характеристиками, стоимостью.
3. Получить удобную возможность правильного подбора автомобильных товаров для своего автомобиля.
4. В случае возникновения каких-либо вопросов иметь возможность задать вопрос менеджеру или заказать обратный звонок.
5. Ознакомиться с порядком оформления заказа, предоставляемых гарантиях, вариантах доставки товара, его возврате и обмене в случае необходимости.
6. Иметь возможность ознакомиться со схемой проезда к торговым точкам предприятия.

1.3.2 Требования к функциям сайта

На веб-сайте предприятия для ИП Коренева И.В. должны быть реализованы следующие функции (исходя из общих требований к сайту):

1. Корзина товаров с автоматическим пересчетом итоговой суммы к оплате (в нее пользователь будет «складывать» товары, представленные на сайте).

2. Модуль подбора шин и дисков по параметрам (ширина, профиль, диаметр, сезон, ценовой диапазон, производитель, в наличии) и модуль подбора шин и дисков по модели автомобиля (марка, модель, год выпуска, модификация).

3. Каталог автомобильных товаров (рубрикатор).

4. Форма оформления заказа для клиента (выбор товаров, изменение их количества, выбор способа оплаты (наличные, кредит, безналичный расчет для юридических лиц), добавление комментария к заказу).

5. Отправить сообщение (чат с менеджером по продажам).

6. Форма заказа обратного звонка (поля для заполнения: имя и телефон).

7. Интерактивная схема проезда к каждой торговой точке (карта Яндекс).

8. Слайд-шоу (демонстрация фотографий торговых точек).

9. Поиск на сайте по введенной фразе.

10. Счетчик посещений (пользователей сайта).

1.3.3 Требования к программному и техническому обеспечению

Поскольку веб-сайт предприятия ИП Коренева И.В. будет размещен для общего доступа в сети Интернет на специальном компьютере – веб-сервере, то возможно определить требования к программному и техническому обеспечению веб-сервера.

Веб-сервер – это сервер, принимающий HTTP-запросы от клиентов (веб-браузеров), и выдающий им HTTP-ответы (HTML-страницы, изображения, файлы). Веб-сервером называют как программное обеспечение, выполняющее функции веб-сервера, так и непосредственно компьютер (аппаратное обеспечение), на котором это программное обеспечение работает. Клиент, которым обычно является веб-браузер, передает веб-серверу запросы на получение ресурсов, обозначенных URL-адресами.

В данном случае, для реализации веб-сайта для ИП Коренева И.В. на компьютере, на котором будет размещен разработанный веб-сайт как минимум должен функционировать веб-сервер типа Apache версии не ниже 1.3.26 под управлением операционной системы типа Windows Server и система управления базами данных MySQL версии не ниже 3.23.

На клиентской стороне под управлением операционной системы типа Microsoft Windows должен быть установлен браузер для полнофункционального просмотра веб-сайта типа Microsoft Internet Explorer версии 7.0 и выше.

Сайт должен быть работоспособен (информация, расположенная на нем, должна быть доступна) при отключении в браузере поддержки JavaScript.

Для функционирования будущего сайта необходимо следующее техническое обеспечение со следующими минимальными характеристиками:

1. Процессор Intel Pentium III с тактовой частотой не ниже 1 ГГц.
2. Оперативная память объемом не менее 512 Мб
3. Свободное дисковое пространство на жестком диске не менее 512 Мб.

1.3.4 Разработка дерева функций

Дерево функций веб-сайта представляет декомпозицию функций системы и формируется с целью детального исследования функциональных возможностей веб-сайта и анализа совокупности функций, реализуемых на различных уровнях иерархии системы (рис. 1.3).

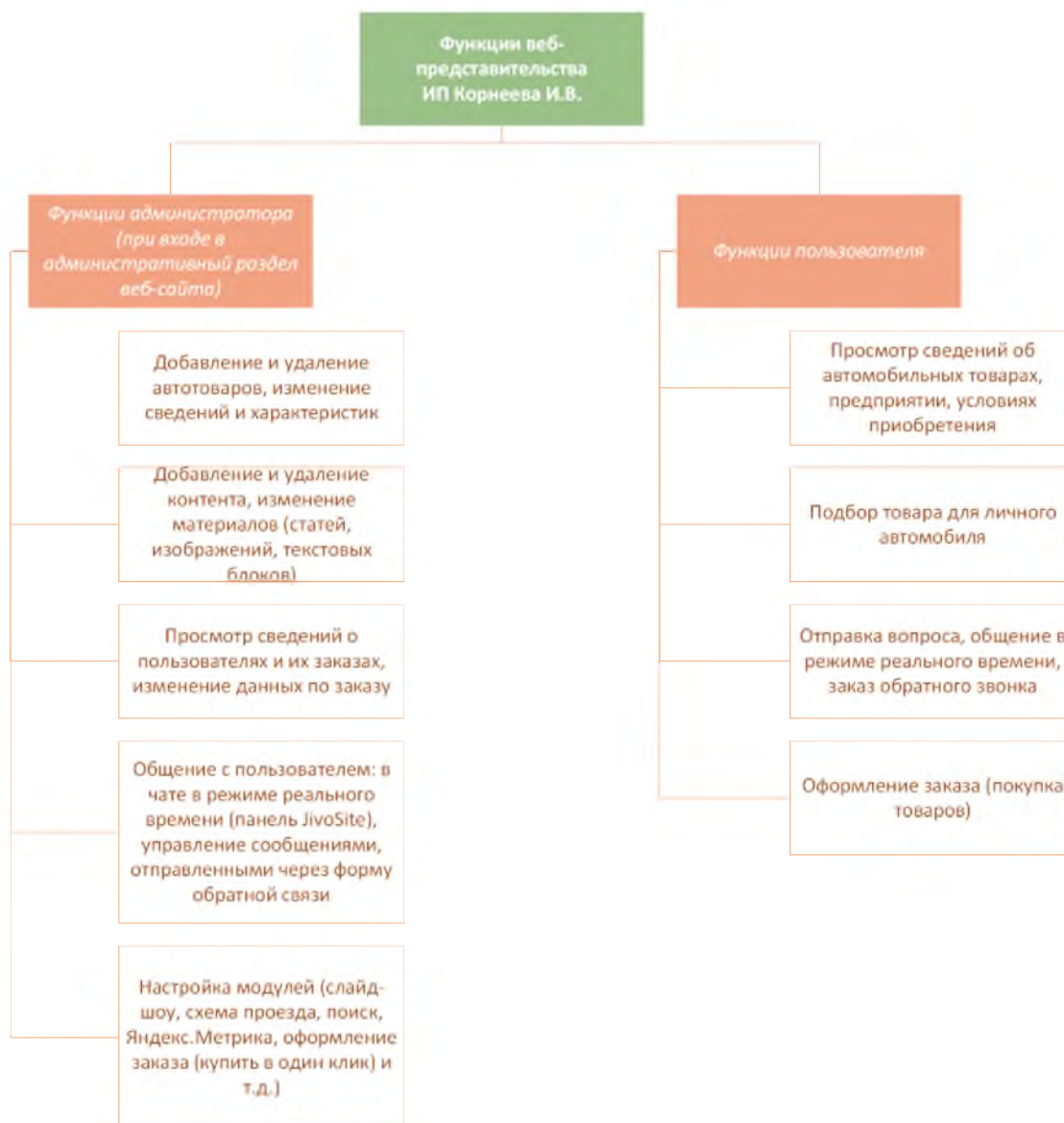


Рис. 1.3 Дерево функций веб-сайта

На базе дерева функций системы осуществляется формирование структуры системы на основе функциональных модулей. В дальнейшем структура на основе таких модулей покрывается конструктивными модулями (для технических систем) или организационными модулями (для

организационно-технических систем).

Этап формирования дерева функций является одним из наиболее ответственных не только при анализе, но и при синтезе структуры системы. Ошибки на этом этапе приводят к созданию «систем-инвалидов», не способных к полной функциональной адаптации с другими системами, пользователем и окружающей средой.

Формирование дерева функций представляет процесс декомпозиции целевой функции и множества основных и дополнительных функций на более элементарные функции, реализуемые на последующих уровнях декомпозиции.

Исходными данными для формирования дерева функций являются основные (функции администратора) и дополнительные (функции пользователя) функции системы.

Дерево функций наглядно демонстрирует разделение функций системы на администраторские и пользовательские.

1.3.5 Разработка сценария диалога

В любой информационной системе должна быть предусмотрена возможность обмена сообщениями между пользователем и вычислительной системой. Такая функция реализуется в виде диалога, поскольку именно диалог является наиболее привычной формой общения для человека. Таким образом, в каждой системе присутствует диалоговая система, а в некоторых случаях вся система может быть сведена к диалоговой системе. Диалоговая система играет важную роль при обработке информации, поэтому требуется особое внимание уделить проектированию и реализации диалоговой системы.

Под диалоговой системой понимается комплекс технических,

информационных и программных средств, который обеспечивает управление диалогом: информирование пользователя, ввод данных и вывод результатов. Особую роль при разработке играет лингвистическое обеспечение.

В дипломе целесообразно привести основные характеристики проектируемой диалоговой системы: является она активной или пассивной, какой тип языка используется для ведения диалога, предусмотрено ли накопление знаний о предметной области и т.д.

Для формализованного описания используются графы, конечные автоматы, и специализированные формально-логические языки.

В данной работе автоматизации подлежит процесс заказа пользователем автомобильных товаров (шин и дисков) посредством веб-сайта, а именно рутинная работа, которую выполняет пользователь сайта и его администратор.

Сценарий диалога пользователя с веб-представительством представлен на рисунке 1.4.

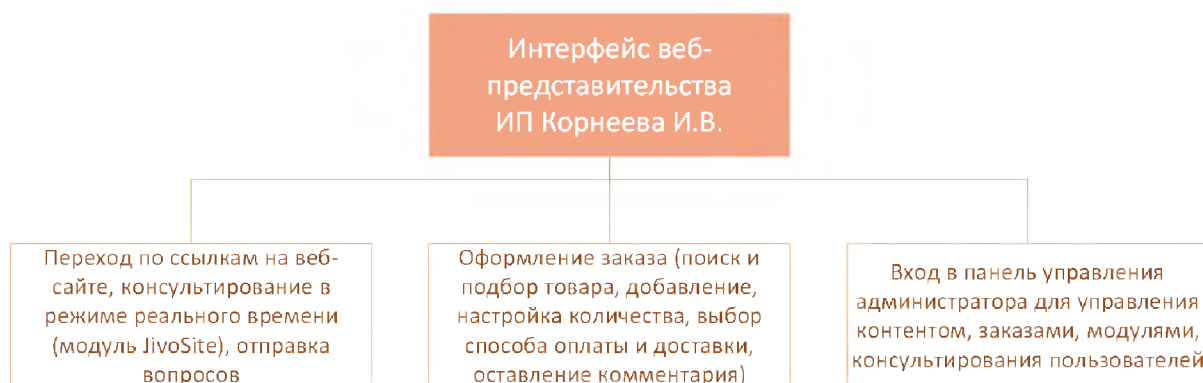


Рис. 1.4 Сценарий диалога веб-сайта

Обмен информацией между пользователем и компьютером (точнее, его программным обеспечением) по всем формальным признакам соответствует понятию «диалог». Развитие диалога во времени можно рассматривать как последовательность переходов системы из одного состояния в другое. Ни одно из этих состояний не должно быть тупиковым, т.е. пользователь должен иметь возможность перейти из любого текущего состояния диалога в требуемое (за один или несколько шагов). Для этого в ходе разработки

интерфейса необходимо определить все возможные состояния диалога и пути перехода из одного состояния в другое – разработать сценарий диалога. Цели разработки сценария: выявление и устранение возможных тупиковых ситуаций в ходе развития диалога; выбор рациональных путей перехода из одного состояния диалога в другое (из текущего в требуемое); выявление неоднозначных ситуаций, требующих оказания дополнительной помощи пользователю.

Сложность разработки сценария определяется в основном двумя факторами – функциональными возможностями создаваемого приложения (т.е. числом и сложностью реализуемых функций обработки информации) и степенью неопределенности возможных действий пользователя. Степень неопределенности действий пользователя зависит от выбранной структуры диалога. Наибольшей детерминированностью обладает диалог на основе меню, наименьшей – диалог типа «вопрос-ответ», управляемый пользователем. Сценарий диалога можно упростить, снизив степень неопределенности действий пользователя. Возможные способы решения этой задачи: использование смешанной структуры диалога (применение меню с целью «ограничения свободы» пользователя там, где это возможно); применение входного контроля вводимой информации (команд и данных).

1.4 Постановка задачи

На основе анализа организационной структуры предприятия была определена цель данной дипломной работы – разработка веб-представительства (Интернет-магазина) для предприятия ИП Коренева И.В., в котором реализуется возможность продажи товара посредством сети Интернет.

Для достижения поставленной цели работы необходимо решить следующие задачи:

1. Выполнить анализ технологий создания веб-сайтов.
2. Обосновать выбор технологий создания веб-сайта.
3. Разработать структуру веб-сайта.
4. Разработать структуры базы данных веб-сайта.
5. Разработать дерево функций и сценарий диалога веб-сайта.
6. Подготовить и согласовать дизайн веб-представительства.
7. Реализовать программные модули веб-сайта.
8. Выполнить тестирование веб-сайта.

В целом для реализации веб-представительства необходимо определить задачи, стоящие перед будущим ресурсом, итоговые ожидания результатов работы. После определения главной цели решается вопрос о структуре веб-сайта. В зависимости от стоящих задач составляется главное меню, обсуждаются элементы навигации и т.д. Затем стоит рассматривать все вопросы бюджета и варианты исполнения поставленной задачи.

Постановка задачи является логическим завершением сбора и анализа информации о проекте который предстоит разрабатывать.

Заказчик веб-сайта: руководство предприятия ИП Коренева И.В.
Название сайта: «Автошина31» (при выборе адреса (доменного имени) в сети Интернет желательно использование слова autoshina31).

Главная цель создания сайта: осуществление продаж посредством веб-сайта, ведение рекламной деятельности и удаленное взаимодействие с клиентами посредством сети Интернет.

Целевая аудитория сайта: автовладельцы и другие контрагенты на территории г. Белгород и России (клиенты предприятия).

2. ПРОЕКТНАЯ ЧАСТЬ

2.1 Описание технологий создания сайтов

Веб-сайт логически можно разделить на две большие части: серверную и клиентскую.

Техническая или серверная часть (back-end) представляет собой файлы, которые содержат исходный код на различных языках программирования (например, PHP, Ruby, Python). Код обрабатывает данные, полученные от пользователя, проверяет данные, и, в зависимости от результатов, может выполнять различные действия. Серверная часть также отвечает за генерацию страниц или различных их участков. К серверной части также относят базы данных (например, MySQL), которые хранят информацию для сайта и настройки. Серверные файлы также отвечают за работу с базой данных, добавляют информацию в сайт, удаляют ее, редактируют, а также выводят в соответствии с запросом клиента. Упрощенно, серверная часть хранит информацию, управляет ей и генерирует клиентскую часть сайта в соответствии с различными условиями. Серверную часть пользователь как таковую не видит в браузере, а видит только результаты ее работы.

Клиентская или пользовательская часть включает все то, что пользователь видит в своем браузере при посещении веб-сайта. Клиентскую часть обеспечивают такие составляющие как HTML, CSS, JavaScript.

Создание сайтов – творческий процесс, а для реализации всех своих идей необходимо владеть современными технологиями, которые в свою очередь постоянно совершенствуются. На сегодняшний день реализовать сайт можно с помощью множества веб-технологий: HTML, CSS, AJAX, PHP, PERL, JSP, ASP, CMS и многие другие. Для разработки сайта с большим набором функций потребуется внедрение дополнительных технологий, например, языка JavaScript, который работает на стороне клиента, т.е. в

браузере пользователя. Огромное количество языков (PHP, ASP) в ходе своей работы (на стороне сервера) генерируют информацию на выходе в формате HTML (на стороне клиента). Каждый из этих языков имеет определенные функциональные особенности, свои недостатки и преимущества. Немаловажную роль при разработке сайтов играют и языки баз данных (например, MySQL, SQL), именно с их помощью работает та или иная база данных, без которых невозможно создавать многофункциональные «динамические» веб-сайты.

Выбор технологий реализации веб-сайта должен максимально отвечать целям создания будущего проекта.

В ходе разработки дизайна разработчик создает несколько проектов, из которых совместно с клиентом выбирают один, способный максимально выгодно представить товар предприятия (в данном случае, автомобильный). Дизайн является ключевым фактором, который формирует первое впечатление о предприятии. Именно за счет визуального воздействия удается расположить к себе потенциальных клиентов. Значение имеет все: цветовая гамма, шрифт и стиль начертания символов, графические элементы, расположение меню и т.д.

По способу реализации веб-сайта предприятия может быть статическим (т.е. представлять собой набор связанных гиперссылками веб-страниц) и динамическим (сайт представляет собой набор веб-страниц, но основной контент отображаемый на веб-странице предварительно записывается, а затем извлекается из базы данных, например, MySQL, рис. 2.1.

Исходя из всего вышесказанного, можно сделать вывод, что разработать качественный и многофункциональный сайт невозможно без знаний нескольких веб-технологий, на которых базируются современные сайты.

В целом технологии разработки сайтов можно разделить на две

группы:

А) «Ручная разработка» с использованием специального программного обеспечения (Adobe DreamWeaver, WebBuilder, SharePoint Designer) для работы с веб-технологиями HTML, CSS, AJAX, PHP. На чистом языке сделаны крупнейшие сайты с посещаемостью в сотни миллиардов пользователей, например, Instagram, YouTube, Pinterest, Dropbox, Twitter, Facebook, Amazon, LinkedIn и другие.



Рис. 2.1 Принцип работы с системой управления содержимым

Б) «Автоматизированная разработка» с использованием систем управления содержимым типа Joomla, WordPress, Drupal. Система управления содержимым (контентом) (от англ. Content Management System) является наиболее простой и функциональной технологией разработки практически любого сайта. Наиболее распространены следующие технологические платформы, используемые в качестве основы веб-приложения, реализующего работу CMS: PHP, Perl и .NET.

Каждый разработчик использует те технологии разработки, которые считает наиболее подходящими для себя и отвечающие целям проекта.

В данном разделе проанализирована деятельность предприятия ИП Коренева И.В., разработана организационная структура предприятия, приведено обоснование целесообразности разработки, а также анализ требований к ней, разработан макет (графическая структура) главной страницы сайта, проведен анализ технологий (способов) создания сайтов.

2.2 Обоснование выбора технологий создания сайта

При выборе технологий для реализации любого проекта, так и в частности для веб-сайта ИП Коренева И.В. необходимо учитывать ряд важных критериев: размер и тип проекта, сложность проекта, скорость разработки, стоимость работы специалистов, доступность специалистов, доступные инструменты разработки, наличие готовых решений, гибкость решения, наличие широкого сообщества, отказоустойчивость решения, тренд его развития, наличие подробной документации, стоимость поддержки, требования к нагрузкам, требования к безопасности, кроссплатформенность, возможности интеграции с другими решениями.

Руководствуясь указанными критериями, разработчик может добиться более объективного выбора технологий разработки сайта и тем самым сэкономить собственное время и денежные средства.

Говоря о сложности разработки проекта веб-сайта для ИП Коренева И.В. можно сказать, что проект представляет собой задачу средней сложности, т.к. веб-представительство требует реализацию не просто сайта-визитки или нетипового решения типа портала или социальной сети, а информационного ресурса с функциональными возможностями Интернет-магазина.

Следует понимать, что сложность проекта влияет на набор используемых технологий, например, при разработке портала может

одновременно использоваться несколько языков программирования. Пример, система Google, ее разные части реализованы на C/C++, Java, Python, JS и других языках. Кроме того, Google активно создает новые технологии, такие как AngularJS.

Приведем краткую характеристику популярных языков:

Язык HTML – язык разметки гипертекста, основа всемирной паутины, имеет собственный набор символов, с помощью которых браузеры отображают страницу. Документ содержит только текст, а другие объекты встраиваются в документ в момент его отображения браузером с помощью тэгов и хранятся отдельно.

Язык PHP – язык используется для создания скриптов, работающих на стороне сервера, позволяет обрабатывать данные HTML-форм, динамически генерировать веб-страницы, создавать машиннонезависимые скрипты, создавать приложения с графическим интерфейсом и т.д. Язык используется в основном для простых и средних проектов. Имеются коробочные решения. С выходом последней седьмой версии, язык получил мощные возможности.

Язык Python – современный язык, разработка на нем быстрая и качественная. Используется для средних и больших проектов.

Язык Ruby – современный язык, разработка на нем также быстрая. В основном используется для разработки простых и средних проектов.

Язык Java – разработка на язык длительная и дорогая. Язык используют в основном для больших проектов со специфическими требованиями.

Язык C# – аналог языка Java, также используют для больших проектов.

Язык программирования JavaScript разработан для создания интерактивных HTML-документов. Объектно-ориентированный язык разработки встраиваемых приложений, выполняющихся как на стороне клиента, так и на стороне сервера. Синтаксис языка похож на синтаксис Java. Основные области применения JavaScript: динамическое создание документа с помощью сценария; оперативная проверка достоверности

заполняемых пользователем полей форм HTML до передачи их на сервер; создание динамических HTML-страниц совместно с каскадными таблицами стилей и объектной моделью документа; взаимодействие с пользователем при решении локальных задач, решаемых приложением JavaScript, встроенном в HTML-страницу. Язык используют для средних и больших проектов, но действительно мощные возможности получил недавно, потому примеров больших проектов пока мало.

Следует упомянуть и фреймворки – некую среду разработки для программистов, где есть готовая инфраструктура и ряд готовых функций со стандартными решениями типичных задач. На каждом языке есть много разных фреймворков. Есть как общие, которые создавались для разработки любых решений, так и специализированные, под узкие задачи. Например, Sylius – специализированный фреймворк для электронной коммерции на основе Symfony. На фреймворках разрабатываются большие и сложные сайты с уникальным функционалом, что значительно быстрее и дешевле, чем разработка проекта на чистом языке. Примеры популярных фреймворков и платформ: PHP: Symfony, Laravel; Python: Django; Ruby: Ruby On Rails; Java: Spring; C#: .NET; JS: Node.js, AngularJS.

Сегодня перечисленные языки широко используются в сфере веб-технологий, но есть много новых языков, которые развиваются, но пока они довольно «сырые», в частности Scala.

Разработка веб-сайта может быть упрощена за счет использования систем управления содержимым или CMS. Самые популярные из них: Joomla, Wordpress, Drupal, Битрикс. Системы по назначению бывают общими, для Интернет-магазинов, для блогов, для форумов и т.д. Разные по условиям использования: платные и бесплатные.

Использование CMS при разработке веб-сайта имеет ряд преимуществ:

1. Снижение расходов (в частности на сопровождение) и простота управления сайтом посредством интерфейса.

2. Широкие возможности для настройки (визуальные редакторы для изменения контента) и поддержка различных расширений для расширения функционала сайта, шаблонов.

3. Поддержка контента в актуальном виде за счет наличия встроенных инструментов наполнения ресурса новым контентом.

4. Администрирование сайта одновременно несколькими пользователями с различными правами доступа.

5. Встроенные средства защиты ресурса от взлома.

Количество систем управления содержимым неуклонно растет. Многие специалисты испытывают затруднения при выборе CMS для своих проектов. В таблице 2.1 приведены сравнительные характеристики четырех популярных систем управления содержимым: Drupal, Битрикс, WordPress, OpenCart.

Таблица 2.1

Сравнение функционала систем управления содержимым

Функциональные возможности системы	Drupal	Битрикс	WordPress	OpenCart
Простота и удобство панели администрирования сайтом	3	4	5	4
Простота и удобство добавления функций Интернет-магазина	2	5	2	4
Возможность интеграции с решениями фирмы 1С (обмен данными с конфигурациями 1С)	2	5	3	4
Визуальный редактор содержимого	3	5	4	4
Поддержка мультимедийного контента	4	5	5	5
Адаптивная верстка	5	4	4	4
Готовность к большим объемам контента	5	5	3	4
Мультиязычность и мультисайтовость	3	4	4	4
Поддержка различного количества модулей	4	5	4	5
Возможность разработки индивидуальных модулей	3	4	5	3
SEO-функционал для поисковой оптимизации	3	4	3	4
Общее количество баллов по всем критериям	37	50	42	45

Оценка функциональных возможностей системы выполнена по пятибальной шкале от 0 до 5 в условных единицах (баллах) на основании отзывов разработчиков веб-сайтов (<https://1ps.ru/blog/sites/cms-overview-1/>) с использованием рассматриваемых систем управления содержимым.

Принцип работы с большинством систем управления содержимым при разработке сайта одинаков: расстановка информационных блоков, прорисовка веб-дизайна, настройка интерфейса. Поскольку изготовление сайтов на базе готовых систем не предполагает разработку собственного движка, цены по такой технологии будут весьма привлекательны.

Система Drupal представляет собой каркас (или CFM – фреймворк управления содержимым) для проектирования систем управления контентом для общего назначения, поставляется со встроенным фреймворком Symfony. Довольно мощная, на ней есть известные сайты, например, официальный сайт Белого Дома. Система бесплатная. Данному движку присущи некоторые минусы: медленная загрузка страниц, большой размер дистрибутива, сложность в освоении. Для стабильной работы требуется надежный выделенный сервер.

Система WordPress – ранее движок для создания блогов, сейчас позволяет создавать практически любые сайты, включая магазины. Одна из самых популярных CMS в мире, есть примеры довольно посещаемых сайтов. На WordPress создаются часто информационные сайты, в том числе СМИ. Система бесплатная.

Система OpenCart – популярная система для небольших и простых Интернет-магазинов в России. Система бесплатная.

Система 1С-Битрикс (рис. 2.2) – одна из самых популярных CMS общего назначения в России. Платформа используется для разработки Интернет-магазинов – надежная и устойчивая к высоким нагрузкам система, соответствующая высоким стандартам безопасности. В настоящее время на платформе «1С-Битрикс: Управление сайтом» работает более 20 000

интернет-магазинов. Возможности системы очень широкие, что позволяет делать большие и сложные сайты. Система интегрируется с прикладными решениями 1С «из коробки». Система платная, но при этом выигрывает за счет технической поддержки пользователя.

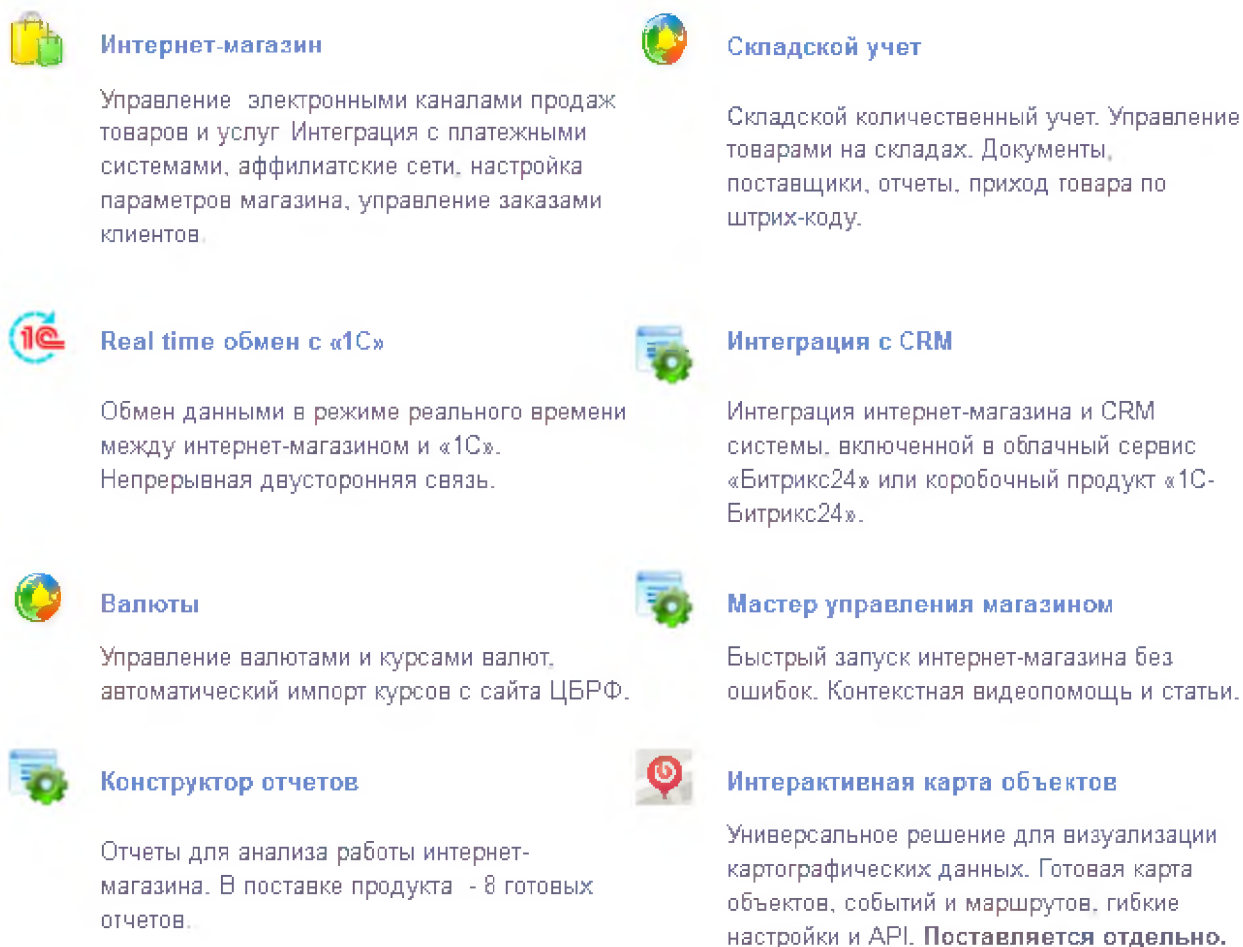


Рис. 2.2 Ключевые возможности 1С-Битрикс

Исходя из общего количества набранных баллов по всем критериям (каждый критерий оценивается по пятибальной шкале) можно заключить, что для реализации веб-сайта ИП Коренева И.В. «Автошина31» система управления содержимым 1С:Битрикс подходит наилучшим образом.

2.3 Построение функциональной модели системы

В работе рассмотрен бизнес-процесс «Анализ требований заказчика

веб-представительства» и построена система с точки зрения разработчика (менеджера проекта), т.к. он отвечает за проект в целом на основании требований заказчика, предоставляемой им информации и контенте, руководствуясь при этом определенными отраслевыми стандартами.

Для анализа основного бизнес-процесса разработки веб-сайта с помощью функционального моделирование построена контекстная диаграмма, которая была в дальнейшем декомпозирована (разбита) на пять основных взаимосвязанных работ (рис.2.3, рис. 2.4).

Контекстная диаграмма по методологии SADT (IDEF0) позволяет провести анализ объекта управления и решаемой задачи. Диаграмма предназначена для описания бизнес-процессов, т.е. устойчивой, целенаправленной совокупности взаимосвязанных видов деятельности (последовательности работ), которая по определенной технологии преобразует входы в выходы, представляющие ценность для потребителя.



Рис. 2.3 Функциональная модель основного бизнес-процесса «Разработка веб-сайта» (контекстная диаграмма)

Методология SADT предназначена для моделирования данных и объектов, а также функционального моделирования. Методология является единственной методологией, легко отражающей такие системные характеристики, как управление, обратная связь и исполнители. Методология SADT началась как язык описания функционирования систем общего вида, а по мере применения ее процедуры описания систем были улучшены и

дополнены.

Разработчики формализуют процесс создания системы, разбивая его на следующие фазы:

1. Анализ – определение того, что система будет делать, проектирование – определение подсистем и их взаимодействие,
2. Реализация – разработка подсистем по отдельности,
3. Объединение – соединение подсистем в единое целое, тестирование – проверка работы системы,
4. Установка – введение системы в действие,
5. Функционирование – использование системы.

Основные положения методологии SADT:

1. Четко сформулированная цель, вытекающая из списков вопросов, ответы на которые мы хотим получить.
2. Модель описывает единственный субъект.
3. Для построения модели выбирается единственная точка зрения.
4. Модель – взаимосвязанные наборы диаграмм.

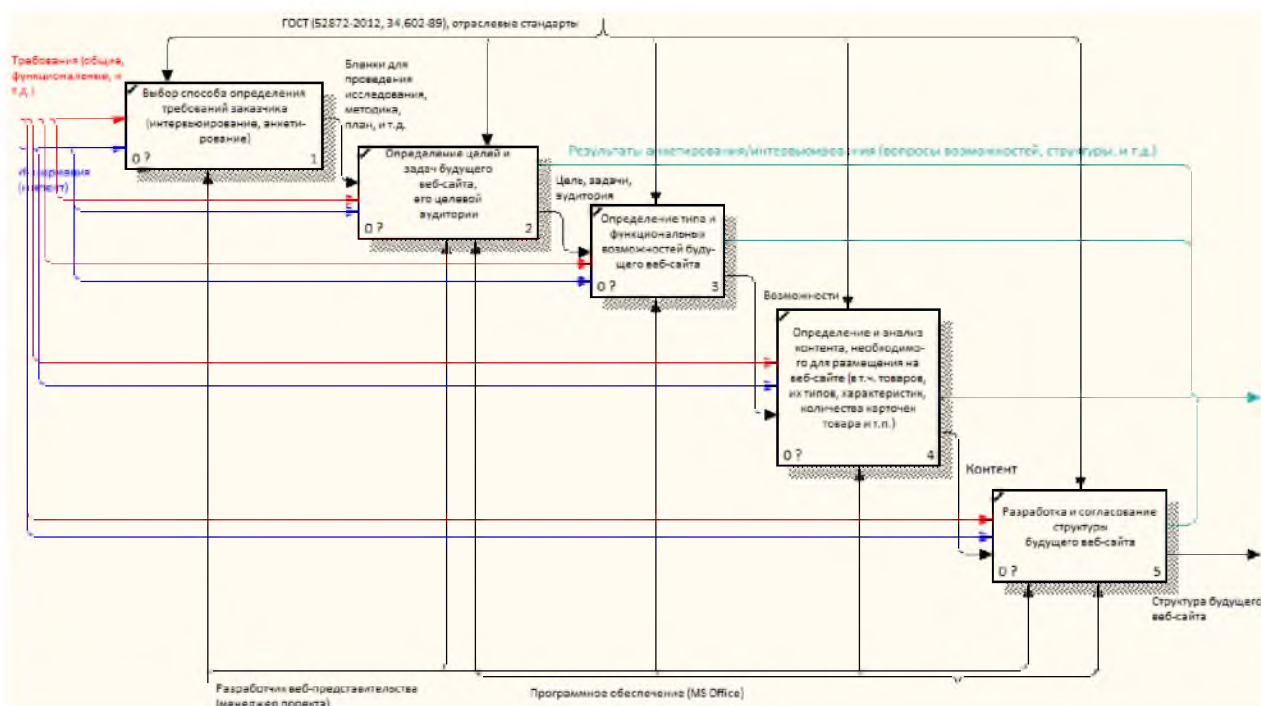


Рис. 2.4 Функциональная модель основного бизнес-процесса «Разработка веб-сайта» (диаграмма декомпозиции)

Модель в данном случае представляет собой описание системы, у которой есть единственный субъект, цель и одна точка зрения.

Диаграмма состоит из:

1. Прямоугольников (функции системы). Каждый блок имеет номер.

Доминирование происходит по времени или по управленческой функции.

2. Дуг (стрелки) – изображают взаимосвязь между объектами.

Стандартная SADT-диаграмма содержит:

1. Слева – вход.
2. Справа – выход.
3. Снизу – механизм.
4. Сверху – управление.

В методологии SADT требуется только пять типов взаимосвязей между блоками для описания их отношений:

1. Управление.
2. Вход.
3. Обратная связь по управлению.
4. Обратная связь по входу.
5. Выход-механизм.

Связи по управлению и входу являются простейшими, поскольку они отражают прямые воздействия, которые интуитивно понятны и очень просты. Отношение управления возникает тогда, когда выход одного блока непосредственно влияет на блок с меньшим доминированием.

Процесс моделирования включает:

1. Сбор информации об исследуемой области.
2. Представление полученной информации в виде модели.
3. Анализ объектов, входов в систему.
4. Анализ функций систем.
5. Построение диаграммы.
6. Уточнение модели по средствам итеративного рецензирования.

Модели создаются с конкретной целью, и эта цель записана на контекстной диаграмме модели (диаграмма верхнего уровня). В каком-то смысле эта цель определяет, как будет использоваться модель. Как только завершено создание модели с требуемым уровнем детализации и модель проверена, она может применяться для достижения поставленной цели.

Проведение интервью с руководителем ИП Коренева И.В. позволило выявить требования к разрабатываемому веб-сайту, который должен быть доступен пользователям по адресу www.autoshina31.ru, его цели, задачи, целевую аудиторию, функциональные возможности.

Выполненный функциональный анализ позволяет тщательно определить порядок разработки веб-сайта, а именно его такую важную составляющую как «информационная структура», анализ также служит основой для четкой схемы действий разработчика. Описание результата анализа – структура сайта – представлено в следующем пункте данной работы.

2.4 Разработка структуры сайта

На основе проведенного устного собеседования с руководителем ИП Коренева И.В. была определена информационная структура веб-сайта, которая включает следующие основные разделы: «Главная», «Шины», «Диски», «Акции», «Кредит», «Покупателю», «Оптом», «Магазины».

Описание информационной структуры сайта:

1. Главная (содержит общие сведения об организации, рекламные баннеры в виде автосменяющегося слайд-шоу, модуль поиска и подбора шин и дисков, блок популярных автотоваров, блок новых автотоваров, блок автотоваров со скидкой, статью);

2. Шины (содержит модуль поиска и подбора шин и дисков);

3. Диски (содержит модуль поиска и подбора шин и дисков);
4. Акции (содержит карточки автотоваров со скидкой);
5. Кредит (содержит сведения об условиях кредитования покупателей);
6. Покупателю (содержит информацию по направлениям: как оформить заказ, оплата, доставка, возврат и обмен, гарантии, написать руководителю).
7. Оптом (содержит информацию для оптовых покупателей автотоваров).
8. Магазины (содержит сведения о местонахождении предприятия, режим работы, контактные данные: адрес, телефон).

Структуру сайта представим на рис. 2.5.

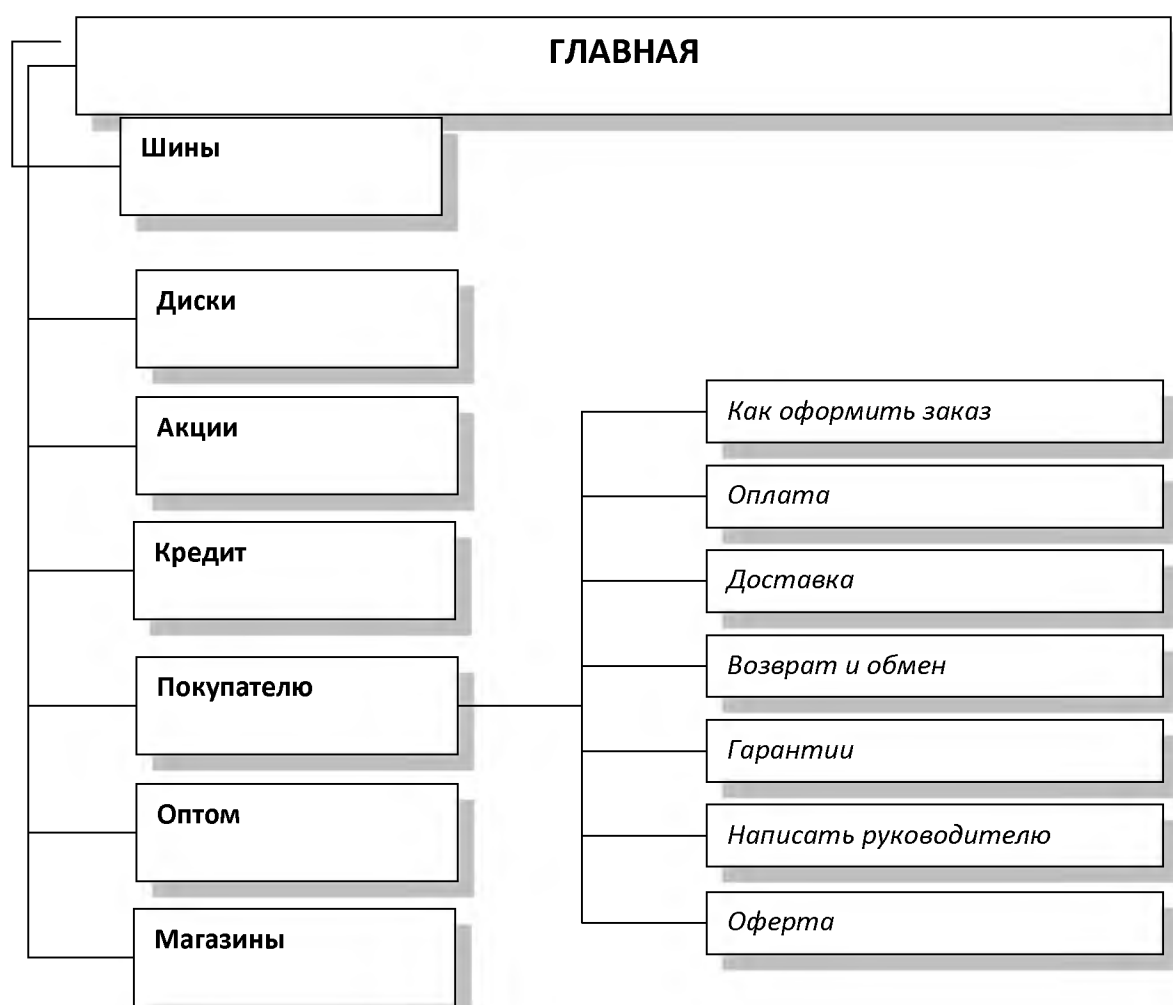


Рис. 2.5 Структура сайта

Для каждого из разделов информационной структуры был получен материал от администрации предприятия, который в дальнейшем подвергся

обработке и сортировке по разделам. Также были подготовлены и отредактированы основные статьи для ресурса. Изначально написание статей производится в текстовом процессоре Microsoft Word.

Графические изображения были отсняты, некоторые их части обрезаны, произведена фотокоррекция яркости и контрастности изображений. Изображения сохранены в графических форматах *.jpg и *.png. Вся обработка изображений производится в программе PhotoShop. Внешнее оформление сайта представлено в приложении (Рис 1-11.)

2.5 Разработка структуры базы данных

В основе веб-сайта для ИП Коренева И.В. лежит база данных MySQL.

База данных необходима для хранения сведений о клиентах, их заказах, характеристик автомобильных товарах (шинах, дисках), их остатках на складах, производителях, совместимости с конкретным автомобилем. Представим схему базы данных на рис. 2.6

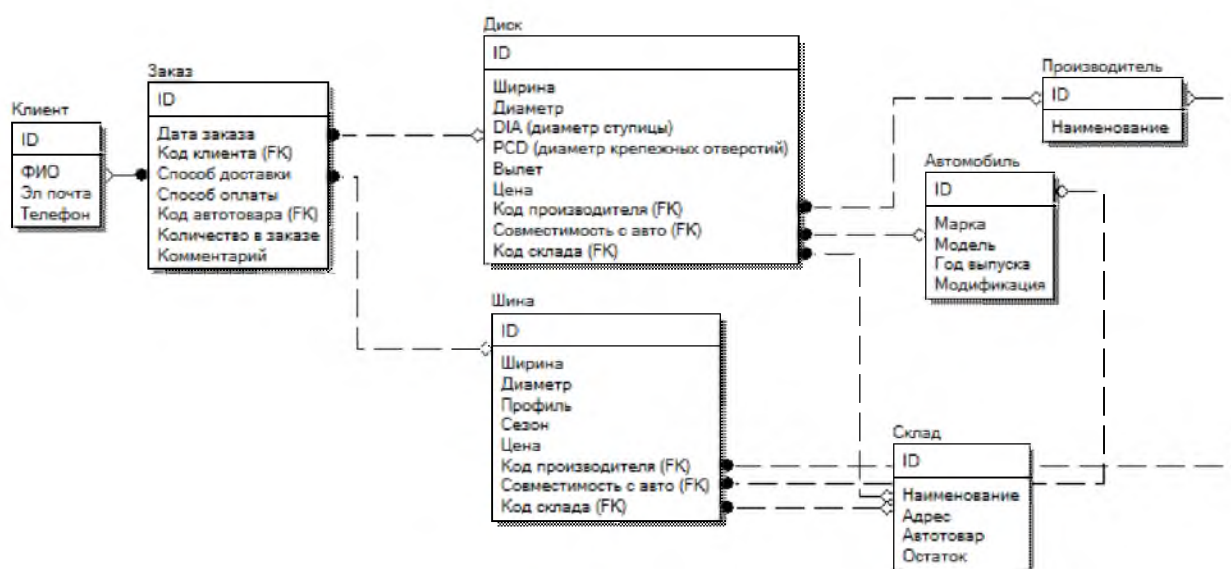


Рис. 2.6 Схема данных базы данных веб-сайта

Подобная структура базы данных позволит посетителю веб-сайта

быстро находить товар по интересующим его характеристикам.

Логическая схема базы данных разработана в приложении ERWin, которое позволяет определять сущности и их атрибуты, устанавливать связи между сущностями (таблицами базы данных), генерировать код базы данных, т.е. структуру базы и данные можно экспортировать в специальные SQL файлы, буфер обмена или на другие серверы, имеется возможность подключения схемы данных к серверу баз данных.

Также необходимо создать в базе данных семь таблиц для хранения соответствующих сведений.

1. Таблица 2.2 для хранения сведений о клиентах (пользователях сайта) имеет следующие поля: код, ФИО, эл. почта, телефон.

Таблица 2.2

Структура таблицы Клиент

Название поля	Ключевое	Тип данных
id	Да	INT(10)
ФИО	Нет	VARCHAR(255)
Эл. почта	Нет	VARCHAR(50)
Телефон	Нет	VARCHAR(50)

2. Таблица 2.3 для хранения сведений о совершаемых пользователями заказах имеет следующие поля: код, дата заказа, код клиента, способ доставки, способ оплаты, код автотовара, количество в заказе, комментарий.

Таблица 2.3

Структура таблицы Заказ

Название поля	Ключевое	Тип данных
id	Да	INT(10)
Дата заказа	Нет	TIMESTAMP
Код клиента	Нет	INT(10)
Способ доставки	Нет	VARCHAR(255)

Способ оплаты	Нет	VARCHAR(255)
Код автотовара	Нет (внешний ключ)	INT(10)
Количество в заказе	Нет	INT(10)
Комментарий	Нет	VARCHAR(255)

3. Таблица 2.4 для хранения характеристик о продаваемых дисках имеет следующие поля: код, ширина, диаметр, DIA, PCD, вылет, цена, код производителя, совместимость с авто, код склада.

Таблица 2.4

Структура таблицы Диск

Название поля	Ключевое	Тип данных
id	Да	INT(10)
Ширина	Нет	FLOAT
Диаметр	Нет	VARCHAR(50)
DIA	Нет	FLOAT
PCD	Нет	VARCHAR(50)
Вылет	Нет	INT(10)
Цена	Нет	FLOAT
Код производителя	Нет (внешний ключ)	INT(10)
Совместимость с авто	Нет (внешний ключ)	INT(10)
Код склада	Нет (внешний ключ)	INT(10)

4. Таблица 2.5 для хранения характеристик о продаваемых шинах имеет следующие поля: код, ширина, диаметр, профиль, сезон, цена, код производителя, совместимость с авто, код склада.

Таблица 2.5

Структура таблицы Шина

Название поля	Ключевое	Тип данных
---------------	----------	------------

id	Да	INT(10)
Ширина	Нет	FLOAT
Диаметр	Нет	VARCHAR(50)
Профиль	Нет	VARCHAR(50)
Сезон	Нет	VARCHAR(50)
Цена	Нет	FLOAT
Код производителя	Нет (внешний ключ)	INT(10)
Совместимость с авто	Нет (внешний ключ)	INT(10)
Код склада	Нет (внешний ключ)	INT(10)

5. Таблица 2.6 для хранения наименований производителей автомобильных товаров имеет следующие поля: код, наименование.

Таблица 2.6

Структура таблицы Производитель

Название поля	Ключевое	Тип данных
id	Да	INT(10)
Наименование	Нет	VARCHAR(255)

6. Таблица 2.7 для хранения сведений о характеристиках автомобилей с целью определения совместимости автотоваров с конкретной моделью автомобиля имеет следующие поля: код, марка, модель, год выпуска, модификация.

Таблица 2.7

Структура таблицы Автомобиль

Название поля	Ключевое	Тип данных
id	Да	INT(10)
Марка	Нет	VARCHAR(50)
Модель	Нет	VARCHAR(50)

Год выпуска	Нет	VARCHAR(4)
Модификация	Нет	VARCHAR(50)

7. Таблица 2.8 для хранения сведений об остатках конкретных товаров на определенном складе предприятия имеет следующие поля: код, наименование, адрес, автотовар, остаток.

Таблица 2.8

Структура таблицы Склад

Название поля	Ключевое	Тип данных
id	Да	INT(10)
Наименование	Нет	VARCHAR(255)
Адрес	Нет	VARCHAR(255)
Автотовар	Нет	INT(10)
Остаток	Нет	INT(10)

Для полей таблиц (столбцов) использованы следующие типы данных: целочисленный – INT, вещественный – FLOAT, символьный переменной длины – VARCHAR, текущая дата и время – TIMESTAMP.

Между таблицами установлены связи (мощность). Например, между таблицами «Клиент» и «Заказ» – установлена связь «один ко многим», т.е. один клиент (посетитель веб-сайта) может разместить множество заказов. Между производителем и автотоваром (диском и шиной), также установлена связь «один ко многим», т.е. один производитель выпускает множество автомобильных товаров (дисков и шин).

В данном разделе приводится обоснование выбора основной платформы разработки веб-сайта (1С-Битрикс, ее сравнение с другими системами управления содержимым), проведен функциональный анализ предметной области (построены модели контекстная и декомпозиции для бизнес-процесса: разработка веб-сайта), приведены расчеты надежности

программного обеспечения (веб-сайта) на основе простой интуитивной модели, разработана структура будущего сайта (определены основные разделы сайта) и база данных, необходимая для его функционирования.

3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

3.1 Описание программных модулей

Система управления содержимым «1С-Битрикс: Управление сайтом» как выбранная в качестве платформы разработки веб-сайта для ИП Коренева И.В. представляет собой профессиональную систему для создания и управления такими Интернет-проектами как корпоративный сайт, Интернет-магазин, информационный портал, Интернет-сообщество, социальная сеть и другим

В данной работе с помощью «1С-Битрикс: Управление сайтом» выполняется разработка нового веб-проекта.



Рис. 3.1 Модули системы управления содержимым «1С-Битрикс»

Систему управления содержимым «1С-Битрикс: Управление сайтом» можно представить в виде 26 модулей, используемых для управления сайтом, а именно контентом, рекламой, Интернет-магазином и многим другим. (рис. 3.1).

Для наглядности взаимодействия программных модулей системы

управления содержимым представим схему. Программное ядро позволяет всесторонне управлять веб-проектами любой сложности.

Архитектура продукта – подход к проектированию, гарантирующий, что программное обеспечение будет отвечать своему предназначению. Архитектура «1С-Битрикс» – это структура, которая включает программные компоненты, видимые снаружи свойства этих компонентов, а также отношения между ними. В основе продукта «1С-Битрикс: Управление сайтом» лежит Bitrix Framework – это созданная на основе PHP платформа для разработки веб-приложений. В отличие от Zend Framework при разворачивании Bitrix Framework разработчик получает не только набор классов, но и развитый интерфейс администрирования.

Структуру Bitrix Framework по уровням архитектуры можно представить в виде групп (рис. 3.2). Фреймворк 1С-Битрикс реализован на файлах, что дает больше свободы разработчику веб-сайта.

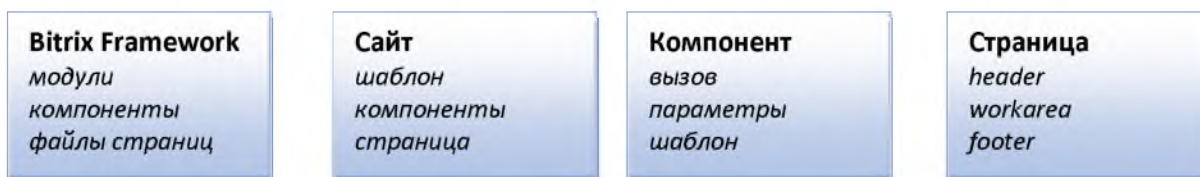


Рис. 3.2 Структура Bitrix Framework по уровням архитектуры

Согласно структуре, модуль фреймворка 1С-Битрикс – это модель данных и API (прикладной интерфейс программирования) для доступа к этим данным. Статические методы классов модуля могут вызываться в компонентах, шаблонах, других модулях. Внутри контекста Bitrix Framework могут создаваться экземпляры классов.

Компонент фреймворка 1С-Битрикс – это контроллер и представление для использования в публичном разделе. Компонент с помощью API одного или нескольких модулей манипулирует данными. Шаблон компонента (представление) выводит данные на страницу.

Страница представляет собой PHP файл, состоящий из пролога, тела страницы (основной рабочей области) и эпилога. Формирование страницы

сайта производится динамически на основе используемого шаблона страницы, данных выводимых компонентами и статической информации, размещенной на странице.

Сайт в понятии Bitrix Framework – это совокупность: учетной записи в базе данных, которая создается в административном меню сайта; публичной части – совокупности скриптов, лежащих в папке сайта и принадлежащих этому сайту; настроек – например, привязка информационного блока к тому или иному сайту, привязка статуса, категории и т.п. к сайту.

Упрощенно набор файлов веб-сайта можно представить в виде схемы (рис. 3.3). Программные модули веб-сайта сохранены в файлах с расширением *.php, а обращение к базе данных MySQL выполняется с использованием запросов. Каждый файл перед выполнением запроса выполняет соединение с базой данных, расположенной на веб-сервере.

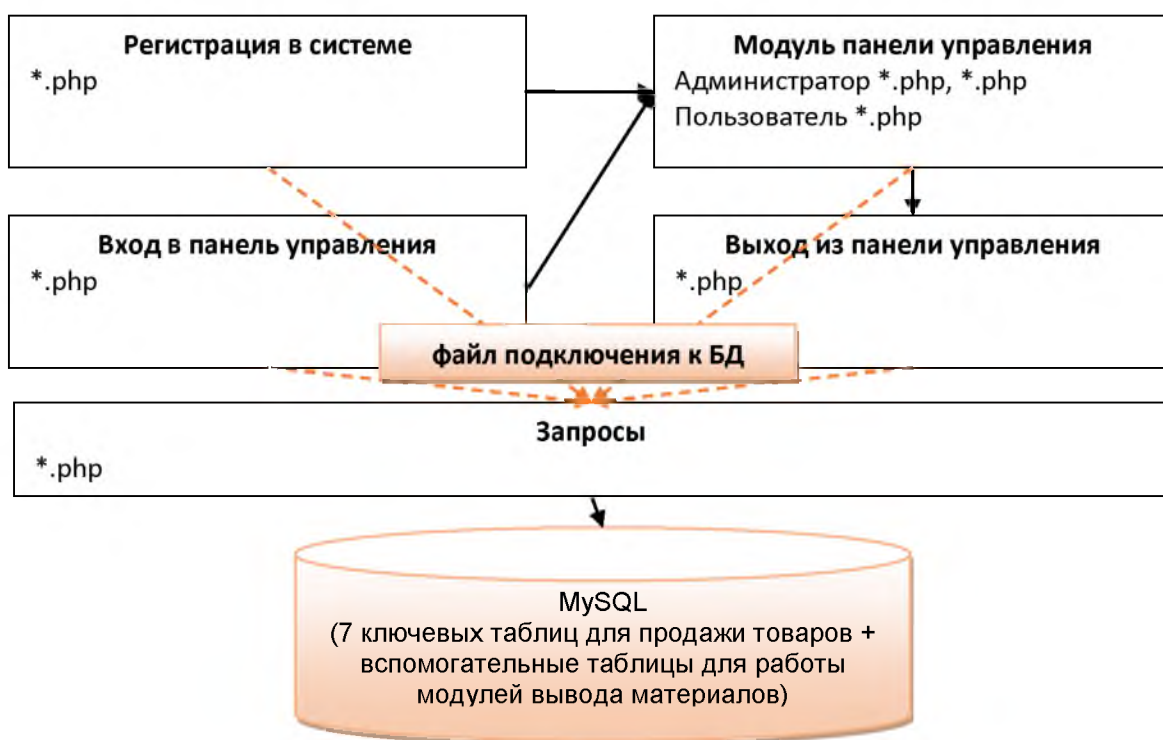


Рис. 3.3 Структурная схема веб-сайта

К программным модулям относятся: модуль регистрации в системе, модуль панели управления для пользователя и администратора, модуль входа в панель управления и модуль завершения работы с панелью управления,

кроме этого модуль подключения к базе данных и модуль запросов к базе данных.

Все файлы проекта веб-сайта были распложены на хостинг-площадке в каталоге зарегистрированного пользователя `/public_html/bitrix/`. В результате проект веб-сайта включает следующие каталоги и файлы:

1. `/admin/` – административные скрипты.
2. `/cache/` – кэш-файлы.
3. `/activities/` – действия для бизнес-процессов.
4. `/components/` – системные и пользовательские компоненты.
5. `/gadgets/` – гаджеты.
6. `/js/` – файлы JavaScript модулей.
7. `/stack_cache/` – кэш-файлы.
8. `/themes/` – темы административного раздела.
9. `/wizards/` – папки мастеров.
10. `/images/` – изображения, используемые системой и модулями.
11. `/managed_cache/` – управляемый кэш.
12. `/modules/` – каталог с модулями системы, каждый подкаталог имеет строго определенную структуру.
13. `/php_interface/` – вспомогательный служебный каталог, включает следующие каталоги и файлы:
 - `dbconn.php` – параметры соединения с базой;
 - `init.php` – дополнительные параметры портала;
 - `after_connect.php` – подключается после соединения с базой;
 - `dbconn_error.php` – подключается при ошибке в момент создания соединения с базой;
 - `dbquery_error.php` – подключается при ошибке в момент выполнения SQL-запроса;
 - `/ID сайта/init.php` – дополнительные параметры сайта;

14. /templates/ – каталог с шаблонами сайтов и компонентов, включает следующие подкаталоги:
- /.default/ – подкаталог с общими файлами, используемыми шаблоном по умолчанию;
 - /autoshina/ – подкаталог с шаблоном сайта;
15. /tools/ – дополнительные страницы, копируемые при установке, которые могут быть использованы на любых страницах сайта.
16. /updates/ – каталог системы обновлений.
17. header.php – стандартный файл, подключающий конкретный пролог текущего шаблона сайта. файл должен использоваться на всех страницах публичной части.
18. footer.php – стандартный файл, подключающий конкретный эпилог текущего шаблона сайта.
19. license_key.php – файл с лицензионным ключом.
20. spread.php – файл, используемый главным модулем для переноса cookies посетителя на дополнительные домены различных сайтов.
21. redirect.php – файл, используемый модулем «Статистика» для фиксации событий перехода по ссылке.
22. stop_redirect.php – файл, используемый модулем «Статистика» для выдачи сообщения посетителю, попавшему в стоп-лист.
23. activity_limit.php – файл, используемый модулем «Статистика» для выдачи сообщения роботу при превышении им лимита активности.
24. rk.php – файл, используемый модулем «Реклама» для фиксации событий клика по баннеру.
25. другие служебные файлы и папки.

Примерный алгоритм действий по созданию веб-сайта следующий:

1. Получение верстки и ознакомление с требованиями руководителя.
2. Определение шаблона и его структуры.
3. Установка дистрибутива «1С-Битрикс».

4. Создание шаблона и применение его к веб-представительству.
5. Создание и настройка необходимых элементов шаблона для оптимизации и продвижения сайта для поисковых машин.
6. Создание структуры сайта.
7. Создание и настройка информационных блоков.
8. Создание шаблонов визуальных компонентов.
9. Тестирование.

Разработка веб-сайта для ИП Коренева И.В. - это многократно повторяющийся процесс, т.е. основана на итеративном подходе к разработке ПО, а именно представляет собой выполнение работ параллельно с непрерывным анализом полученных результатов и корректировкой предыдущих этапов работы: планирование – реализация – проверка – оценка.

3.2 Дизайн веб - представительства

До начала непосредственной подготовки конечного дизайна в графическом приложении типа Adobe Photoshop рекомендуется выполнить разработку прототипа (или эскиза) главной страницы веб-сайта (рисунок 3.4).

После создания прототипа выполняется выбор макета и дизайна главной страницы сайта, а уже затем разработка шаблона. Веб-дизайн – это прежде всего разработка интерфейса, среды взаимодействия пользователя с информацией, а не просто «красивая картинка». Необходимо учитывать особенности веб-среды, понятия «удобство использования», направленность на цели создания сайта. Важно учесть основные сценарии поведения пользователя, особенности целевой аудитории.

Дизайн главной страницы сайта выполнялся в программе Adobe PhotoShop Creative Cloud – многофункциональном графическом редакторе. Продукт является лидером рынка в области коммерческих средств

редактирования растровых изображений и наиболее известным продуктом фирмы Adobe. Несмотря на то, что изначально программа была разработана как редактор изображений для полиграфии, в данное время она широко используется и в веб-дизайне.

Логотип «Автошины у Ирины»	Контактные телефоны заказать звонок	Адреса магазинов	Корзина купить в кредит	
Горизонтальное меню Шины Диски Акции Кредит Покупателю Оптом Магазины			Поиск на сайте	
Слайд-шоу Рекламные баннеры				
Модуль подбора шин и дисков по параметрам и автомобилю Ширина Профиль Диаметр Сезон Цена (от, до) Производитель и т.д.				
Популярные шины				
Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее
Новые поступления шин				
Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее
Шины со скидкой				
Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее	Название Параметры ФОТО Цена Подробнее
Заголовок статьи				
Текст статьи...				
Подвал 2012-2016 © ИП Коренева И.В. «Автошины у Ирины» Счетчик посещений Яндекс.Метрика				
Чат с менеджером «Отправьте нам сообщение» JivoSite				

Рис. 3.4 Прототип (эскиз) главной страницы сайта

Дизайн (макет) сайта веб-сайта представляет собой прототип веб-сайта, в данном случае его отдельной главной страницы, т.е. графический образ с прорисованными деталями различных уровней. Дизайн дает представление о структурном наполнении веб-сайта, навигации и контенте (рис. 3.5).

Дизайн веб-сайта в PhotoShop был начат с разработки концепции. Предварительно было продумано и согласовано с руководством ИП Коренева И.В. размещение элементов на главной странице.

Разработка веб-сайта представляет собой многоэтапную работу. Один из наиболее важных этапов этой работы – дизайн. После того как дизайн утвержден заказчиком, проводилась верстка PSD-макета. Далее производится

верстка, согласно разработанного макета. Важно оптимизировать макет для верстки, т.е. выполнить несколько простых правил:

1. Дизайн веб-сайта отличается от дизайна полиграфической продукции (буклетов, визиток, баннеров и т.д.). Макет должен разрабатываться в традиционном формате (.psd), необходимо использовать цветовое пространство RGB (не CMYK), единицами измерения должны быть пиксели (не миллиметры или другие величины), следует помнить о разрешении экранов популярных на сегодня устройств (в том числе мобильных: планшетов, телефонов и т.д.).

2. Слои должны быть упорядочены, т.е. сгруппированы в смысловые блоки с осмысленными названиями в папки. Желательно придерживаться иерархии сверху вниз и слева направо. Невидимые слои следует удалять, либо объединять (иногда не сливать).

3. Использовать четкие размеры и отступы, т.е. направляющие PhotoShop или готовые модульные сетки, чтобы точно определить расположение элемента на странице. Осознано выбирать пропорции и размеры объектов и делай их кратными, например, 1000px, а не 998px. Размер шрифта должен быть указан целыми числами без дробей. Шрифт не следует предварительно трансформировать (растягивать).

4. Не использовать способы наложения, т.е. конечный PSD-макет не должен содержать визуальных эффектов, достигнутых режимами наложения слоев. Не следует спешить и растривать слои и их эффекты – т.к. в дальнейшем при верстке часто требуется определить параметры градиента, теней или скругления углов.

5. Демонстрировать состояния всех интерактивных элементов, например, кнопок, элементов меню: нормальное, активное, не активное, при наведении, при клике и т.д. Определять все параметры оформления текста: заголовки, списки, таблицы, контент и т. д.

6. Исходные материалы макета следует сохранять в отдельной папке в исходном качестве и дополнительные материалы, которые используются в макете, включая специальные шрифты, на поиск которого может уйти большое количество времени. Давать пояснения, к элементам, вызывающим вопросы. Идеи, которые непросто показать в макете следует записать в отдельный текстовый файл.

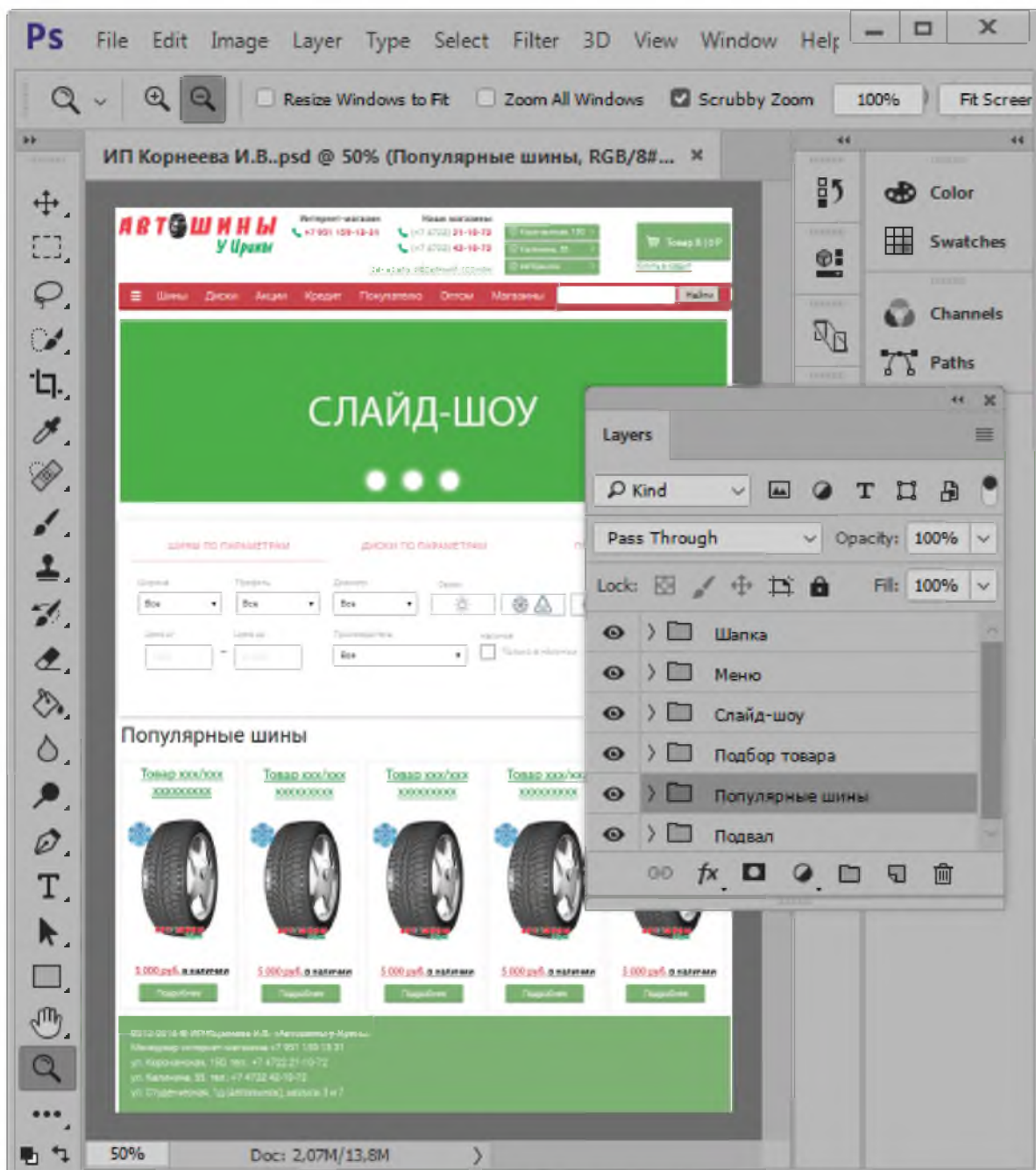


Рис. 3.5 Разработка дизайна главной страницы веб-сайта

Соблюдение рассмотренных правил ускоряет процесс верстки, который выполняется в дальнейшем.

Слои в конечном файле для удобства навигации в процессе верстки сгруппированы в группы: шапка (верхняя часть), меню, слайд-шоу, подбор товара, популярные шины, подвал (нижняя часть). В результате получена многослойная раскладка, полностью готовая к верстке и кодированию. Макет второстепенных страниц сайта отличается информационным наполнением в соответствии с названием каждого раздела.

После разработки дизайна в программе Adobe PhotoShop была выполнена его верстка под систему управления содержимым 1С-Битрикс, другими словами выполняется технологический процесс по интеграции дизайна в систему управления – верстка – перевод из графического PSD-эскиза в код HTML, т.е. создается и программируется макет сайта.

Использование шаблона позволяет проводить гибкую настройку дизайна веб-сайта, разделов и страниц веб-сайта. Например, возможно использование специального дизайна в течение указанного периода времени, автоматическое управление внешним видом сайта в зависимости от группы посетителей и т.д. У сайта может быть много шаблонов дизайна, и, наоборот, один шаблон может быть использован на нескольких сайтах.

Шаблон дизайна – это внешний вид сайта, определяющий расположение различных элементов на веб-сайте, художественный стиль и способ отображения страниц, включает программный HTML- и PHP-код, графические элементы, таблицы стилей, дополнительные файлы для отображения контента, дополнительно может включать шаблоны компонентов, готовых страниц и сниппеты.

Bitrix Framework имеет возможность создания шаблона с помощью встроенного визуального редактора. Однако редактор разработан для работы с контентом, поэтому верстка встроенным редактором не используется.

В данной работе на всех веб-страницах фактически меняется только контентная часть, а дизайн – не изменяется. Другими словами, дополнительные условия в шаблоне сайта не используются.

Проанализировав шаблон было определено, какая часть кода относится к «Прологу» (файл header.php, шапка), какая к «Эпилогу» (файл footer.php, подвал), а какая часть – к «Рабочей области страницы». Выбранные части кода были размещены в соответствующих файлах, а рабочая область была отмечена тегом #WORK_AREA# в шаблоне веб-сайта. Все изображения, относящиеся к шаблону, были размещены в папке /templates/autoshina/images/. Описания стилей из представленной верстки сохранены в файл: /templates/autoshina/styles.css. Описания стилей собственно шаблона сохранены в файл /templates/autoshina/template_styles.css.

В результате верстки дизайна из файла PSD был получен программный код шаблона для платформы 1С-Битрикс. Программный код шаблона был сохранен в каталоге веб-хостинга /public_html/bitrix/templates/autoshina/, где autoshina – идентификатор шаблона сайта, т.е. ID-поле в форме редактирования шаблона сайта (административный пункт меню «Шаблоны сайта»).

Полученная структура программного кода шаблона:

1. header.php – пролог шаблона сайта.
2. footer.php – эпилог шаблона сайта.
3. styles.css – стили шаблона на языке CSS.
4. menu_template.php – шаблон вывода меню соответствующего типа.
5. chain_template.php – шаблон вывода навигационной цепочки.
6. /components/ – каталог с компонентами, принадлежащими модулю.
7. /lang/ – языковые файлы, принадлежащие как данному шаблону в целом, так и отдельным компонентам.
8. /images/ – каталог с изображениями данного шаблона сайта.

9. /page_templates/ – каталог с шаблонами страниц и их описанием, хранящимся в файле content.php.

10. вспомогательные файлы, входящих в шаблон.

Далее в шаблон интегрируются компоненты, т.е. в дизайне веб-сайта были выделены блоки, которые содержат динамическую информацию (вместо них размещается вызов компонентов) и блоки, информация в которых должна изменяться пользователем без изменения шаблона (это включаемые области).

Функциональные области веб-сайта представлены на рисунке 3.6.



Рис. 3.6 Функциональные области веб-сайта

В зависимости от того, где должен быть расположен компонент в файле header.php или footer.php, удаляется HTML-код, отображающий данную функциональную область и вставляется код компонента. Шаблон редактировался без визуального редактора системы.

Компоненты в коде страницы были подключены с помощью функции IncludeComponent().

```
<?APPLICATION->IncludeComponent (
    "bitrix:main.include",
    "",
    Array(
        "AREA_FILE_SHOW" => "file",
        "PATH" => $APPLICATION->GetTemplatePath("include_areas/autoshina.php"),
        "EDIT_TEMPLATE" => ""
    )
);?>
```

Рис. 3.7 Код вставки компонента включаемая область

Код для вставки компонента «Включаемая область» (bitrix:main.include), с настройкой на вывод из файла представлен на рисунке 3.7. В данном случае «include_areas/autoshina.php» представляет собой файл с включаемой областью. В файле включаемой области не требуется подключать пролог и эпилог (файлы header.php и footer.php). Необходимо проверить, что файл включаемой области подключен из системы, а не вызван напрямую.

```
<?APPLICATION->IncludeComponent (
    "bitrix:menu",
    "horizontal_multilevel",
    Array(
        "ROOT_MENU_TYPE" => "top",
        "MAX_LEVEL" => "3",
        "CHILD_MENU_TYPE" => "left",
        "USE_EXT" => "Y",
        "MENU_CACHE_TYPE" => "A",
        "MENU_CACHE_TIME" => "3600",
        "MENU_CACHE_USE_GROUPS" => "Y",
        "MENU_CACHE_GET_VARS" => Array ()
    )
);?>
```

Рис. 3.8 Код вставки меню для шаблона веб-сайта

Для подключения меню необходимо вставить в нужное место шаблона вызов компонента (рис. 3.8).

Далее проведена кастомизация (настройки) шаблона компонента с целью приведения формы вывода данных компонента в соответствие с дизайном веб-сайта.

Для изменения системного шаблона компонента под веб-представительство, его необходимо сначала целиком скопировать в папку шаблона сайта. После выполнено редактирование скопированного шаблона. Например, для редактирования шаблона на основе компонента меню необходимо отредактировать в HTML-верстке код, отвечающий за показ меню: заменить стиль CSS у блока и у списка элементов меню, добавить условие проверки активного пункта меню и первого пункта меню.

После кастомизации шаблона, реализуется полная структура веб-сайта согласно выделенным с руководителем ИП Коренева И.В. разделам: «Главная», «Шины», «Диски», «Акции», «Кредит» и т.д.

Структура сайта представлена в таблице 3.1.

Таблица 3.1

Структура веб-сайта

Раздел сайта	Каталог/файл	Тип информации	Используемые компоненты
Главная	/index.php	Страница с динамической информацией	bitrix:news.list bitrix:catalog.top
Автотовары (шины, диски)	/catalog/shiny/ /catalog/diski/	Страница с динамической информацией	bitrix:catalog
Магазины (карта сайта, служебная информация)	/kontakti/	Статическая страница	bitrix:main.map
Результаты поиска (служебная страница)	/catalog-search/index.php	Страница с динамической информацией	bitrix:search.page
Обработчик ошибки 404	/404.php	Страница с динамической информацией	bitrix:main.map
Обработчик ошибки 500	/500.html	Статическая HTML страница (не Bitrix Framework)	нет

Вывод динамической информации из базы данных в Bitrix Framework осуществляется в основном с помощью информационных блоков. При

разработке веб-сайта была определена структура информационных блоков. Рассмотрим пример использования информационного блока на примере каталога товаров. Реализуемая схема каталогов представлена на рисунке 3.9 (например, группа 1 – шины, группа 1.1 – летние шины, группа 1.2 – зимние шины и т.д.).

- Группа 1
 - Группа 1.1
 - Свой фильтр по свойствам
 - Группа 1.2
 - Свой фильтр по свойствам

Рис. 3.9. Схема каталога товаров веб-сайта

Способ реализации информационного блока: все товары в одном информационном блоке. Информационный блок расположен на первом уровне (Группа 1). Преимущества реализации: иерархия, которой можно управлять из 1С; легко управляемая структура каталогов в рамках сайта. Недостатки реализации: сложности со свойствами товаров, если товары разнородные (в данном случае, практически не критично); свойства будут храниться в одной таблице, что плохо повлияет на производительность.

После выбора схемы реализации определен тип информационного блока, заданы его свойства и выполнено наполнение контентом через импорт данных из информационной базы 1С ИП Коренева И.В. (также возможен импорт из форматов CSV, XML, либо ручное заполнение карточек товаров).

Тщательно проработанная информационная структура и дизайн сайта позволяют повысить удобство работы посетителя с информационным ресурсом.

В целом процесс разработки веб-сайта можно описать следующим образом: разработка дизайна в виде файла PSD, создание (верстка) шаблона, установка шаблона и применение его к веб-сайту, далее создание и настройка необходимых элементов шаблона, в дальнейшем создание структуры,

создание и настройка информационных блоков, создание шаблонов визуальных компонентов и конечное тестирование.

3.3 Тестирование

Диаграмма развертывания моделирует физическое развертывание артефактов на узлах. В данном случае, чтобы описать веб-сайт ИП Коренева И.В. диаграмма развертывания должна показывать, какие аппаратные компоненты («узлы») существуют (например, веб-сервер и сервер базы данных), какие программные компоненты («артефакты») работают на каждом узле (например, веб-приложение, база данных), и как различные части этого комплекса соединяются друг с другом (взаимодействуют посредством удаленного вызова процедур, использования API, веб-сервисов и т.п.).

Узлы представляются как прямоугольные параллелепипеды с артефактами, расположенными в них, изображенными в виде прямоугольников. Один узел диаграммы развертывания может концептуально представлять множество физических узлов, таких как кластер серверов баз данных. Существует два типа узлов: узлы устройств – физические вычислительные ресурсы для выполнения программного обеспечения (ПК, мобильные устройства); узлы среды выполнения – программный вычислительный ресурс, который работает внутри внешнего узла и который предоставляет собой сервис, выполняющий другие исполняемые программные элементы.

В данном случае диаграмма развертывания веб-сайта представляет интерес по двум причинам: показывают архитектуру и организацию файлов (рис.3.10).

После завершения работ над веб-сайтом не следует торопиться размещать его в сети Интернет на хостинг-площадке, поскольку первоначально необходимо проверить как сайт будет выглядеть с точки зрения пользователя, т.е. провести тестирование сайта. Впрочем, такие проверки необходимо проводить не только на этапе первоначального создания веб-сайта, но и после внесения любых (или, по крайней мере, существенных) изменений в содержание сайта. А для такого тестирования следует иметь соответствующие средства.

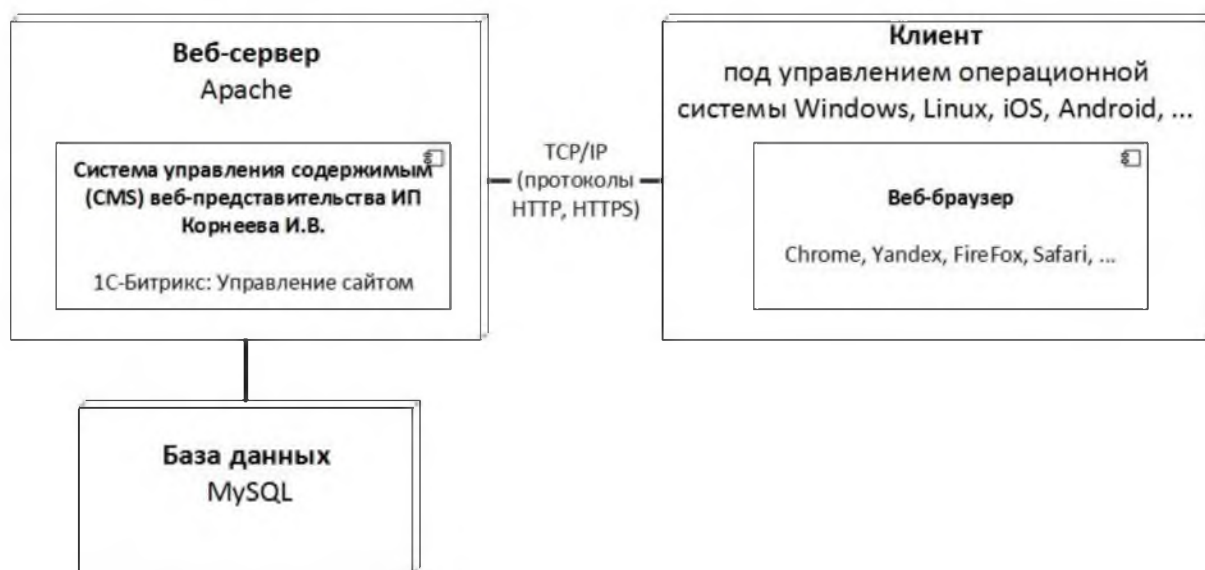


Рис. 3.10 Диаграмма развертывания веб-сайта

Изначально на сайте проверяется система гиперссылок (наличие неработающих ссылок). Чтобы выполнить проверку на наличие битых ссылок был выполнен проход по всем ссылкам сайта. Также можно воспользоваться автоматизированным решением типа инструментария веб-мастера Яндекс и Google (поисковые системы предоставляют отчет по найденным битым ссылкам). Третий вариант использовать специализированное программное обеспечение, проверяющее сайт на битые ссылки, например, Semonitor или Xenu's Link Sleuth. Последний вариант это Интернет-сервисы по проверке битых ссылок, например validator.w3.org/checklink.

Второй этап тестирования – это проверка сайта в разных браузерах. После развертывания сайта на локальном сервере требуется посмотреть, как будут отображаться страницы веб-сайта ИП Коренева И.В. в браузере, т.к. пользователи могут заходить на сайт из различных браузеров (их насчитывается более 60). При этом разные браузеры могут отображать одну и ту же страницу по-разному. Поэтому желательно проверить как сайт будет смотреться если уж не во всех, то хотя бы в самых распространенных версиях браузеров.

Третий этап проверки – соответствие страниц веб-стандартам. Поэтому после завершения верстки сайта выполнена проверка, в какой степени выполнены данные требования. Следует понимать, что браузеры тоже не всегда следуют требованиям стандартов и способны отображать страницы, которые стандартам не соответствуют. Но даже если браузер успешно отображает страницу, не соответствующую стандартам, проблемы могут возникнуть при просмотре страницы роботом поисковой системы. Роботы поисковиков гораздо требовательнее в вопросе соответствия страниц сайта стандартам и могут не проиндексировать страницу, если она выполнена с отклонениями от требований. Проверить страницу на соответствие стандартам можно с помощью сервисов HTML, CSS, RSS, Atom, KML. Удовлетворить всем требованиям (устранить все недостатки, отмечаемые валидатором) очень сложно и существует небольшое количество сайтов, удовлетворяющих всем требованиям.

Четвертый этап проверки – тестирование скорости работы сайта. Тестирование сайта бывает необходимо и после того, как сайт запущен и уже работает достаточно долгое время. На этом этапе бывает иногда даже необходимо посмотреть, как быстро пользователь получает доступ к сайту. Если у пользователя каждая страница открывается по несколько десятков секунд, то достичь высокой посещаемости сайта не удастся.

Для тестирования скорости открытия страниц сайта использован сервис `host-tracker.com`. Сервис позволяет протестировать скорость загрузки страниц с 98 различных точек по всему миру.

Для расчета надежности веб-сайта использована простая интуитивная модель, которая предполагает проведение тестирования веб-сайта (в данном случае двумя программистами) независимо друг от друга, использующими независимые тестовые наборы. В процессе тестирования каждый разработчик фиксируют все найденные им ошибки.

Допустим, если первая группа разработчиков обнаружила n_1 ошибок, а вторая n_2 , тогда n_{12} – это число ошибок, обнаруженных как первой, так и второй группой. Тогда через N обозначим неизвестное количество ошибок, присутствующих в программе до начала тестирования. Эффективность тестирования можно определить, как описано в формулах:

$$E_1 = \frac{n_1}{N} \quad (3.1)$$

$$E_2 = \frac{n_2}{N} \quad (3.2)$$

Эффективность тестирования можно интерпретировать как вероятность того, что ошибка будет обнаружена. Таким образом, можно считать, что группы обнаруживают ошибку в программе с вероятностью:

$$p_1 = \frac{n_1}{N} \quad (3.3)$$

$$p_2 = \frac{n_2}{N} \quad (3.4)$$

где:

p_1 – вероятность обнаружения ошибки первой группой;

p_2 – вероятность обнаружения ошибки второй группой.

Тогда вероятность p_{12} того, что ошибка будет обнаружена обеими группами, можно принять равной:

$$p_{12} = \frac{n_{12}}{N} \quad (3.5)$$

С другой стороны, так как группы действуют независимо друг от друга, то $p_{12} = p_1 * p_2$. Получаем:

$$\frac{n_{12}}{N} = \frac{n_1}{N} * \frac{n_2}{N} \quad (3.6)$$

Отсюда получаем оценку первоначального числа ошибок программы:

$$N = \frac{n_{12}}{E_1 * E_2} \quad (3.7)$$

В процессе тестирования веб-сайта согласно простой интуитивной модели первая группа нашла 6 ошибок, вторая группа нашла 5 ошибок. Рассчитаем надежность в соответствии с формулами, приведенными выше. После исправления в программном коде веб-сайта найденных ошибок две группы вновь приступили к тестированию веб-сайта и поиску новых ошибок и так далее. Результаты тестирования веб-сайта по простой интуитивной модели занесены в таблицу 3.2.

Таблица 3.2

Результаты тестирования веб-сайта двумя программистами

i	n1.i	n2.i	n12.i	E1.i	E2.i	Ni
1	6	5	5	1 1/5	1	6
2	5	5	6	5/6	5/6	8,64
3	3	2	4	3/4	1/2	10
4	1	3	3	1/3	1	9
5	3	3	4	3/4	3/4	7
6	3	4	4	3/4	1	5
7	3	2	3	1	2/3	4
8	1	2	2	1/2	1	4
9	1	2	2	1/2	1	4
10	1	1	1	1	1	1

На рисунке 3.11 изображен график зависимости количества предполагаемых ошибок N от номера теста i. По графику можно сделать вывод о зависимости количества тестов и количества предполагаемых

ошибок – изначально на веб-сайте имеется какое-то количество ошибок, которое уменьшается с каждым последующим тестом и устранением найденных ошибок. В итоге мы приходим к тому, что все собственные ошибки программного обеспечения найдены и устранены.

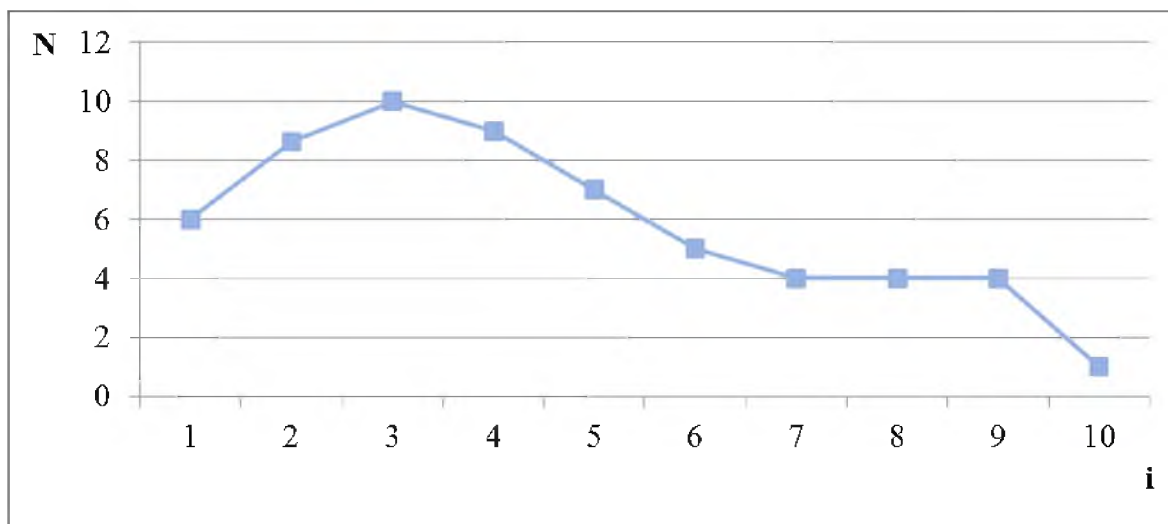


Рис. 3.11 Зависимость количества ошибок N от количества тестов

Проведение тестирования сайта под различные требования является обязательным этапом на пути сдачи проекта заказчику. Ознакомиться с результатом работы можно в приложении (внешний вид сайта предприятия ИП Коренева И.В.).

В данном разделе подробно описан функционал и сценарий диалога пользователя с веб-сайтом, описаны программные модули веб-сайта, Bitrix Framework, порядок разработки дизайна и его верстки, порядок настройки модулей веб-сайта, проведено заключительное тестирование разработки, проведены расчеты согласно простой интуитивной модели. Разработанное и протестированное веб-представительство для ИП Коренева И.В. доступно по адресу www.autoshina31.ru в сети Интернет.

ЗАКЛЮЧЕНИЕ

В ходе написания данной работы выполнена разработка веб-представительства (веб-сайта) для предприятия ИП Коренева И.В. с использованием системы управления контентом «1С-Битрикс: Управление сайтом».

На разработанном веб-сайте для предприятия, который представляет собой информационный Интернет-магазин, клиенты могут быстро ознакомиться с перечнем автомобильных товаров, их характеристиками, стоимостью, получить возможность правильного подбора товаров для личного автомобиля, получить сведения об оформлении заказа, гарантиях, вариантах доставки, возврате и обмене товаров, осуществить оформление заказа и его оплату, написать вопрос менеджеру, заказать обратный звонок, ознакомиться со схемой проезда к торговым точкам и т.д.

Веб-представительство – это оптимальная рекламная площадка, т.е. средство проведения рекламных кампаний и акций в сети Интернет. Предполагается, что в дальнейшем заказчик оптимизирует сайт для поисковых систем, т.е. осуществит «продвижение» и «раскрутку» сайта с целью привлечения большего количества посетителей и как следствие расширит базу своих клиентов. Для этого возможно составить семантическое ядро сайта, создать карту сайта sitemap.xml, настроить файл robots.txt, запустить платную рекламу типа Яндекс.Директ, создать аккаунты для продвижения в социальных сетях и т.д.

Основная цель работы: разработка веб-представительства для предприятия ИП Коренева И.В. достигнута благодаря выполнению поставленных задач:

1. Проведен анализ предметной области: анализ деятельности ИП Коренева И.В., построена организационная структура предприятия. Обоснована целесообразность разработки, выполнен анализ требований

заказчика (общих, функциональных, к программному и техническому обеспечению). Разработана графическая структура веб-представительства. Описаны способы и технологии создания веб-сайтов.

2. Разработан проект веб-представительства: обоснован выбор технологии разработки на базе платформы 1С-Битрикс, выполнено сравнение с другими системами управления содержимым (Drupal, WordPress, OpenCart). Построена функциональная модель бизнес-процесса «Разработка веб-представительства» и построена система с точки зрения разработчика (контекстная диаграмма и диаграмма декомпозиции). Разработана математическая модель с использованием модели надежности программного обеспечения (веб-сайта). Разработана структура сайта. Разработана база данных веб-представительства (включая схему данных на логическом уровне).

3. Разработано веб-представительство: построено дерево функций и сценарий диалога пользователя с веб-сайтом. Описаны программные модули 1С-Битрикс, структура Bitrix Framework. Разработан дизайн веб-представительства в Adobe PhotoShop. Выполнена верстка шаблона из PSD-макета в программный код на HTML, CSS, PHP. Построена диаграмма развертывания веб-представительства. Проведено конечное тестирование веб-сайта.

Разработанное и протестированное веб-представительство для ИП Коренева И.В. доступно по адресу www.autoshina31.ru в сети Интернет.

Внедрение веб-сайта для предприятия ИП Коренева И.В. позволило расширить клиентскую базу, вести продажи автомобильных товаров онлайн, наладить информационное взаимодействие с базой клиентов, вести поддержку клиентов в режиме реального времени, а также проводить рекламные кампании и акции в сети Интернет с целью коммерческой выгоды. Разработанный веб-сайт достаточно удобен в обслуживании, что подтверждается возможностями административной панели 1С-Битрикс:

Управление сайтом. Интерфейс веб-сайта www.autoshina31.ru доступен для ознакомления в приложении к работе.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Басс, Л. Архитектура программного обеспечения на практике / Л. Басс, П. Клементс, П. Кацман. – 2-е изд. – СПб. : Питер, 2006. – 575 с.
2. Божко, В.П., Информационные технологии в экономике и управлении / В.П.Божко – Москва, 2006 – 766 с.
3. Веллинг, Л., Разработка веб-приложений с помощью PHP и MySQL / Л. Веллинг - Вильямс, 2010. – 428 с.
4. Вендров, А.М. Проектирование программного обеспечения экономических информационных систем / А.М. Вендров. – электрон. учебник.
5. Гагарина, Л.Г. Технология разработки программного обеспечения / Л.Г. Гагарина: учеб. пособие.- М. : Инфра-М, 2012. – 623 с.
6. Даккет, Д., HTML и CSS. Разработка и дизайн веб-сайтов / Д. Даккет, Эксмо, 2013.
7. Дейт, К., Введение в системы баз данных / К. Дейт - : Пер. с англ. – М.: Издательский дом «Вильямс», 2005.
8. Седерхольм Д. Пуленепробиваемый веб-дизайн. Библиотека специалиста / Д. Седерхольм, Питер, 2012.
9. Клименко, Р., Веб-мастеринг на 100% / Р. Клименко, Питер, 2013. 615 с.
10. Кузнецов С.Д. Проектирование и разработка корпоративных информационных систем / С.Д. Кузнецов. – электрон. учебник.
11. Курс по разработке веб-сайтов на 1С-Битрикс: dev.1c-bitrix.ru/learning/course/?COURSE_ID=34&INDEX=Y.
12. Мак-Дональд, М., Создание Web-сайта. Недостающее руководство / М. Мак-Дональд, БХВ-Петербург, 2013.
13. Методология SADT, infman.ru/online/sadt.

14. Методология структурного анализа и проектирования SADT, vernikov.ru/biznes-modelirovanie/metodologiya/item/210-sadt-metodology-structurnogo-projectirovanija.html.
15. Никсон, Р., Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript и CSS / Р. Никсон, Питер, 2013. – 456 с.
16. Официальный учебный курс Microsoft. Microsoft Office PowerPoint 2003 – Эком, Бином. Лаборатория знаний, 2006. – 392 с.
17. Программа моделирования бизнес-процессов СА AllFusion Process Modeler (ранее BPWin), ca.com.
18. Соммервил, И., Инженерия программного обеспечения. / И. Соммервил - Пер. с англ. – М.: Издательский дом «Вильямс» 2002.
19. Сравнение CMS omniweb.ru/articles/chto-vybrat-joomla-ili-drupal.
20. Тестирование сайта, website-ru.net/txt/instrument/testirovanie-saita.shtml.
21. Хоган, Б. Книга веб-программиста. Секреты профессиональной разработки веб-сайтов / Ю. Хоган - Питер, 2013.
22. Чепмен, Н., Цифровые технологии мультимедиа / Н. Чепмен 2-е издание: Пер. с англ. – М.: Издательский дом «Вильямс», 2006. – 624 с.

ПРИЛОЖЕНИЕ

Внешний вид разработанного веб-представительства для ИП Коренева И.В. (публичная часть и административный раздел) представлен на рисунках 1-11.

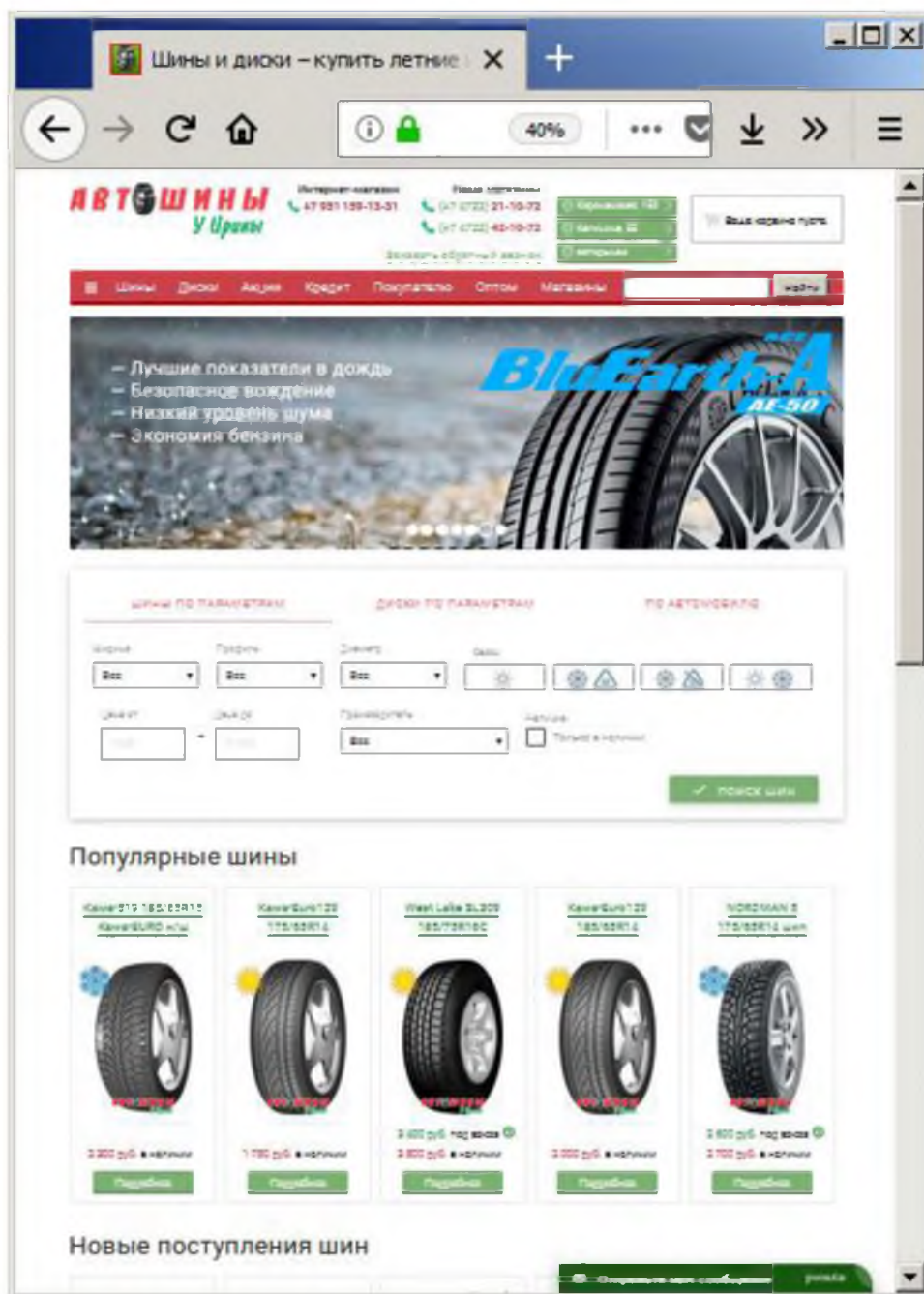


Рис. 1 - Главная страница (верхняя часть, включая модуль подбора товара)

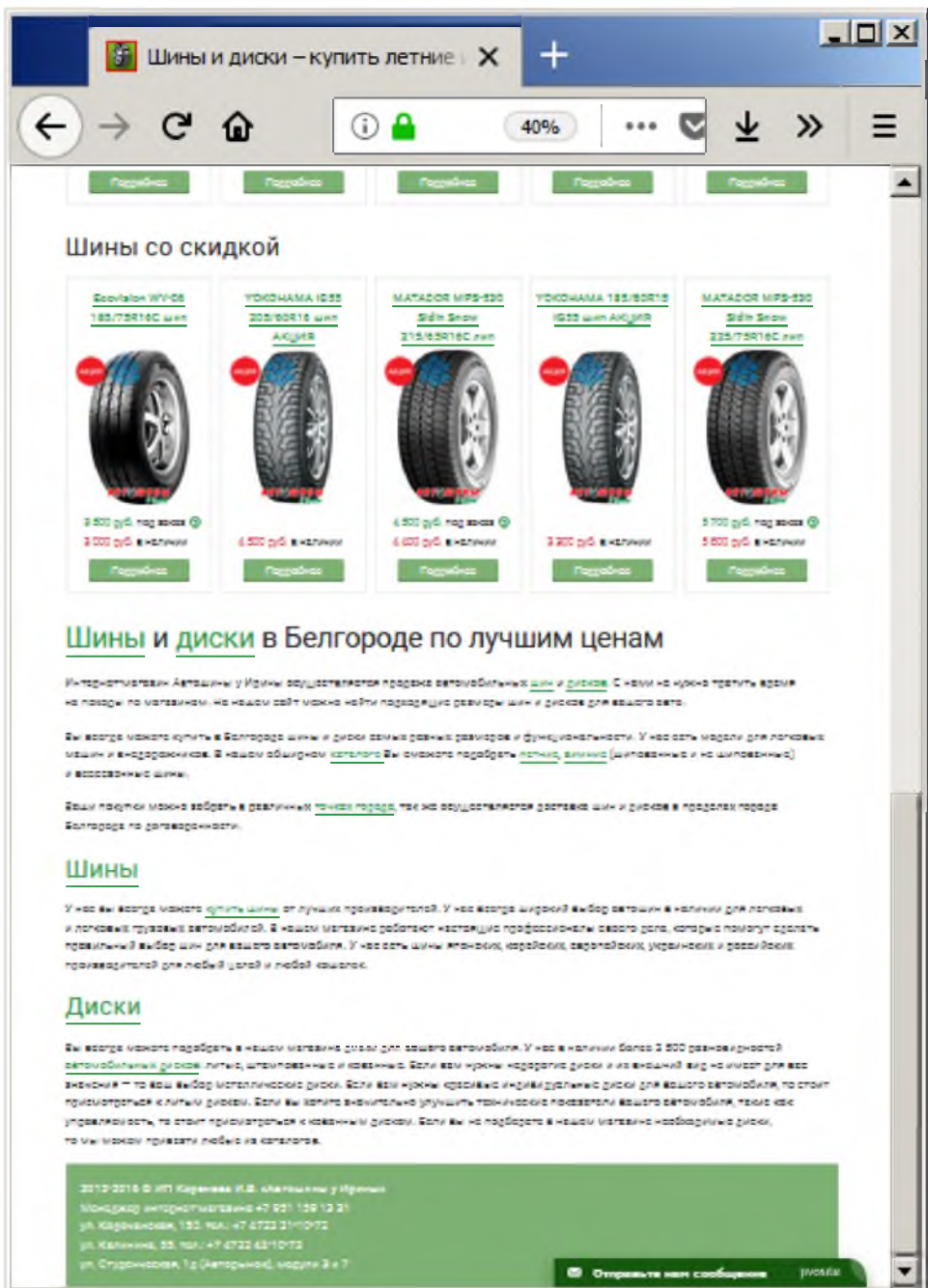


Рис. 2 - Главная страница (нижняя часть)

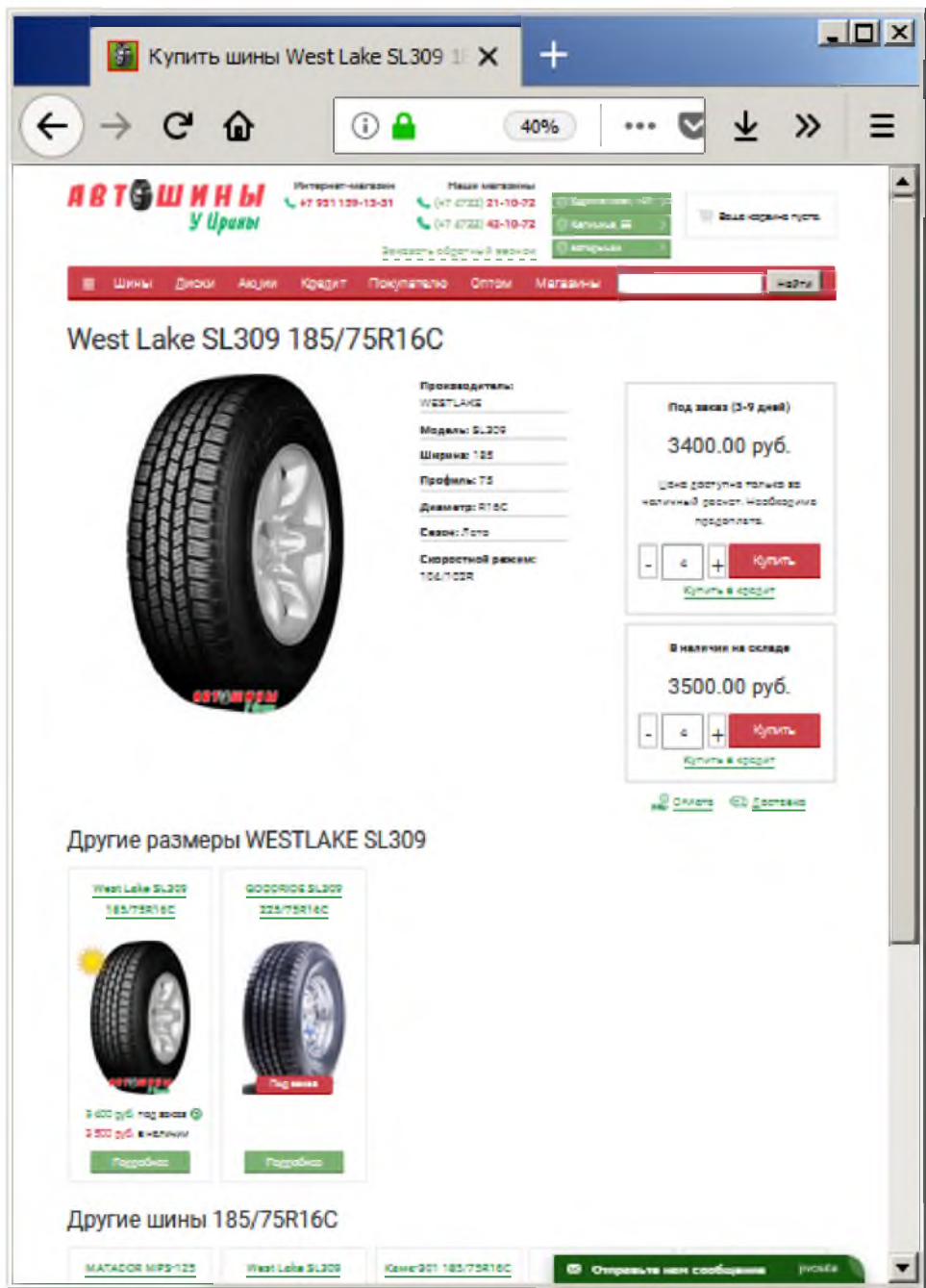


Рис. 3 - Карточка приобретаемого товара

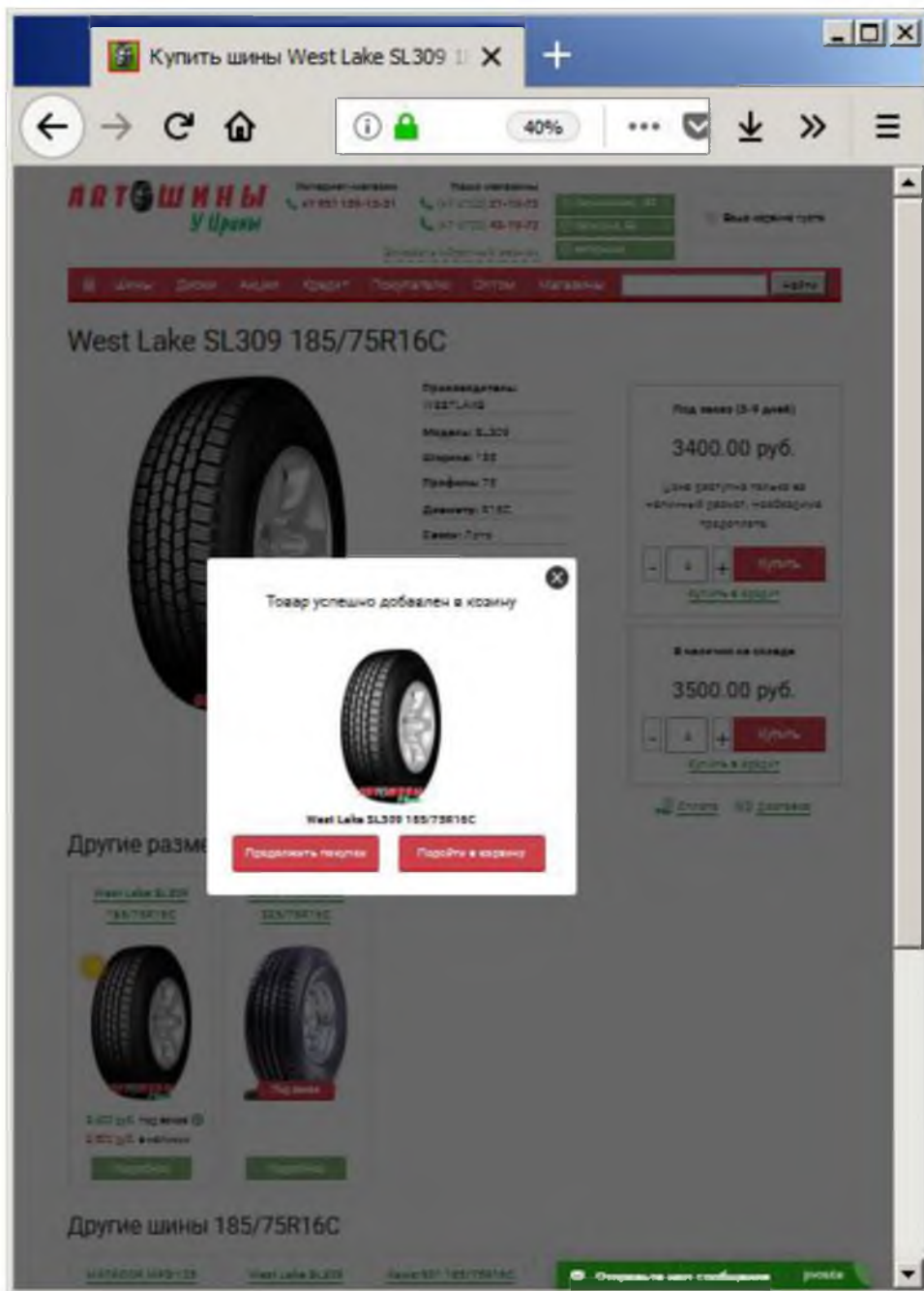


Рис. 4 - Всплывающее окно после нажатия на кнопку «Купить»

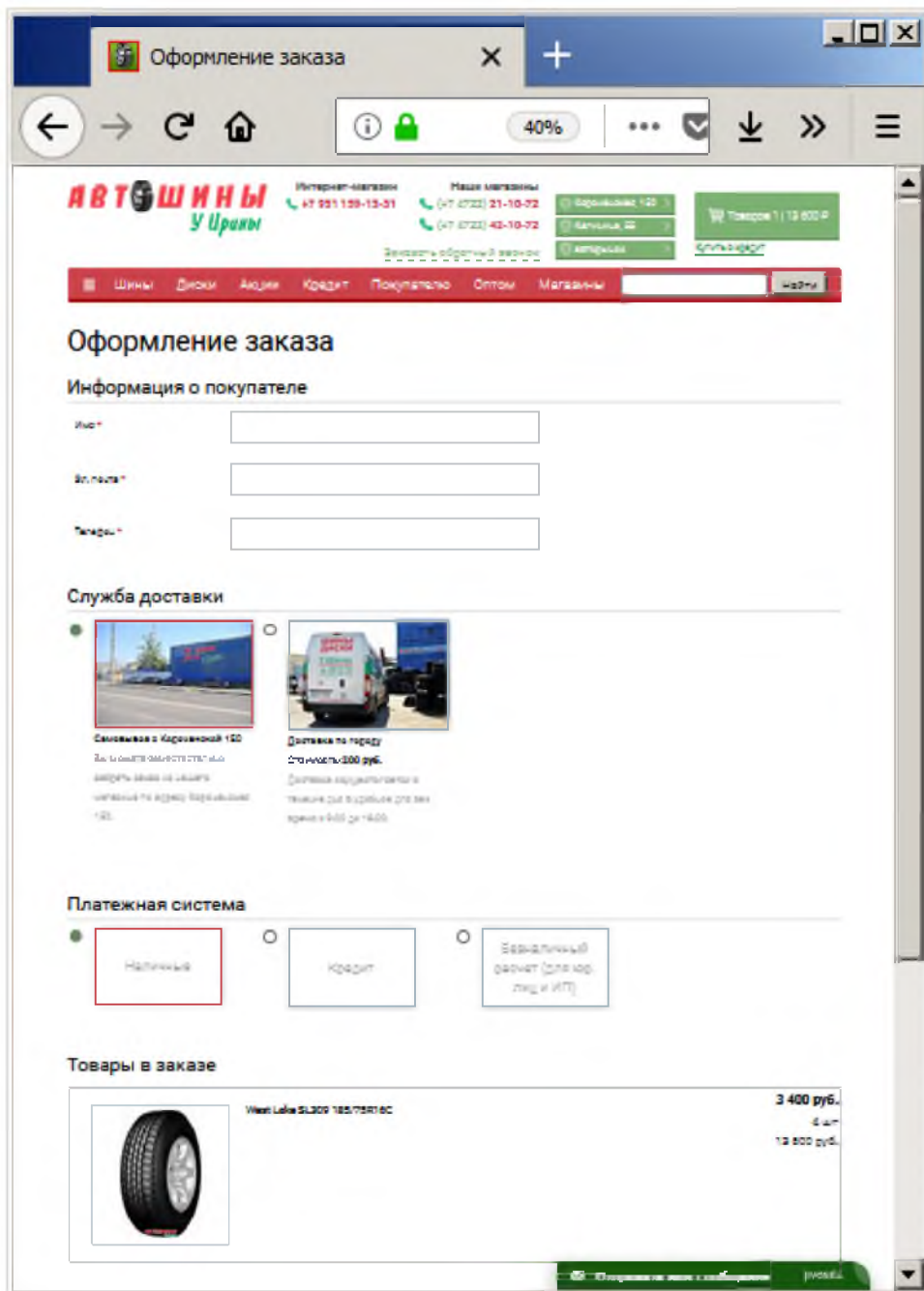


Рис. 5 - Оформление заказа пользователем (имя, эл. почта, телефон, способ доставки, способ оплаты, товар в корзине на сумму, комментарий)

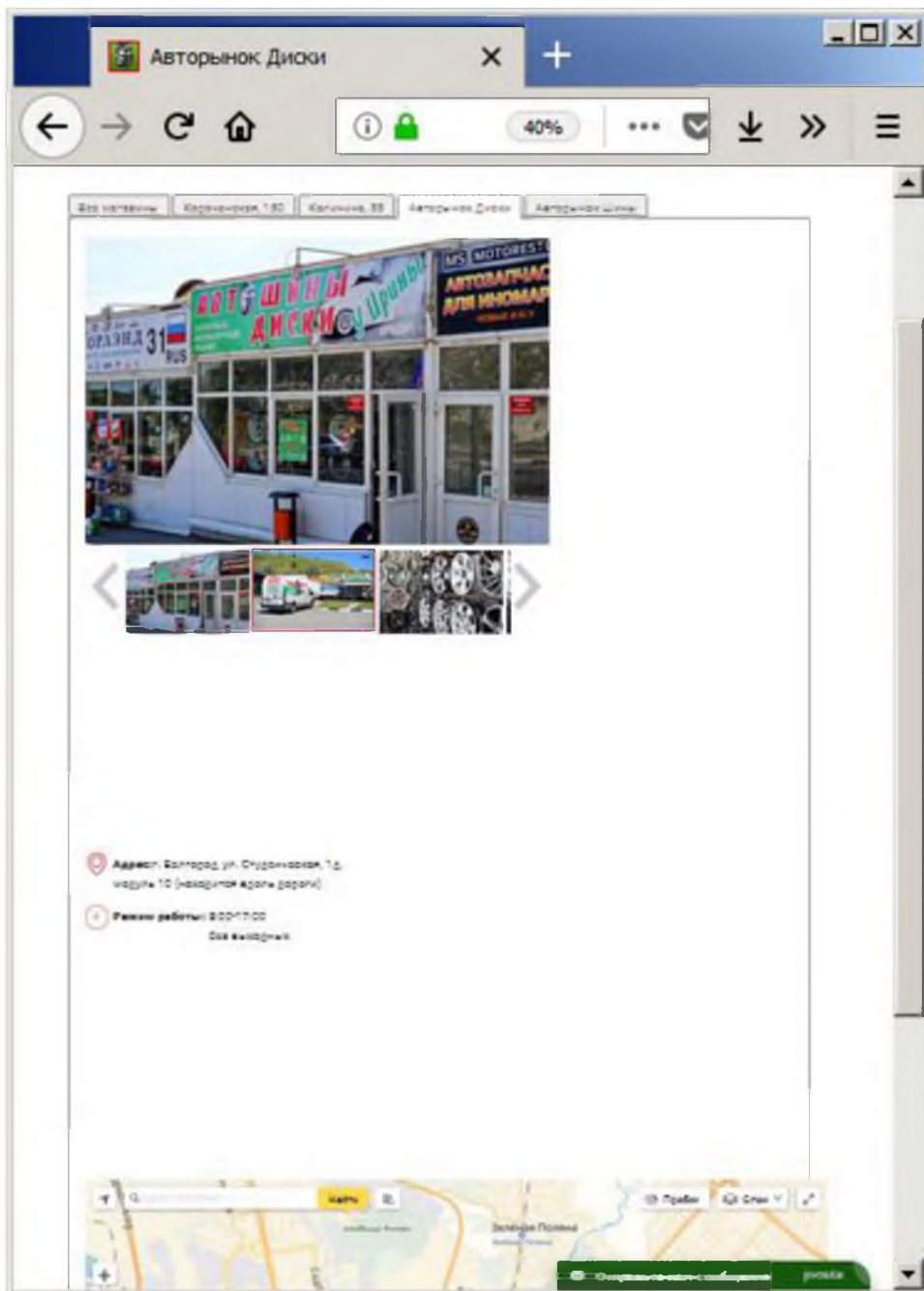


Рис. 6 - Раздел «Магазины» (слайд-шоу и карта Яндекс)

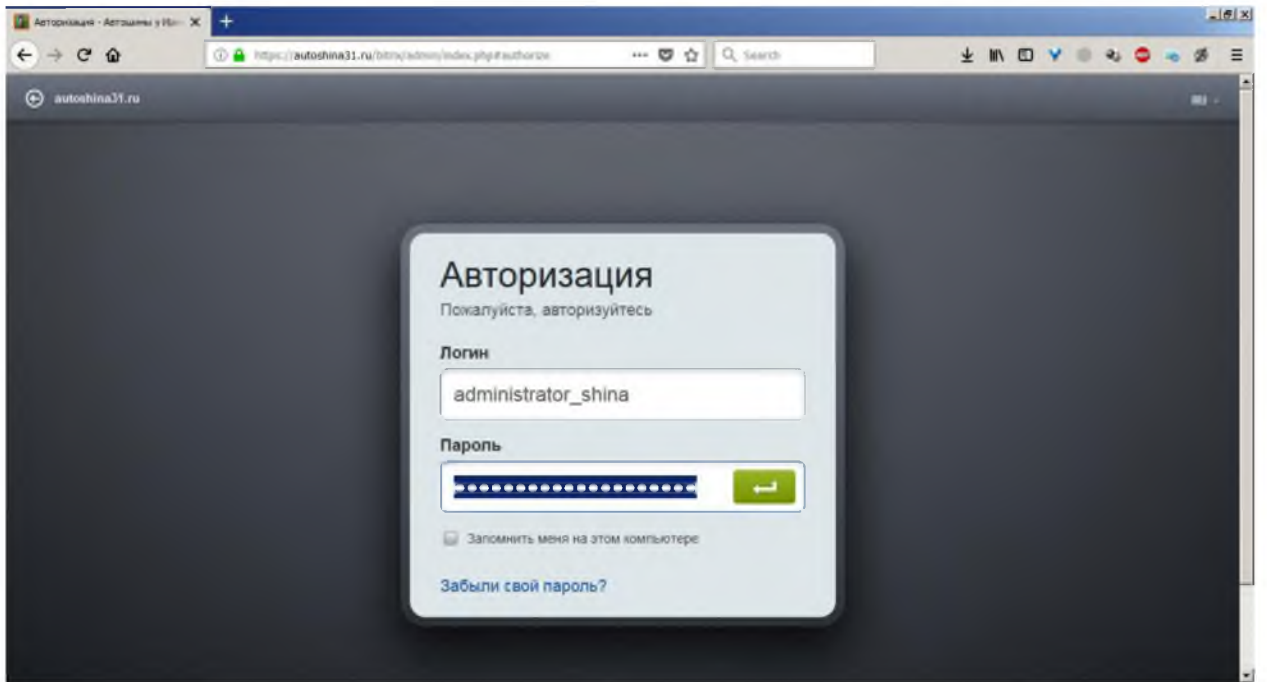


Рис. 7 - Вход в административный раздел сайта
(<https://autoshina31.ru/bitrix/admin/index.php#authorize>)



Рабочий стол – инструмент для организации удобного управления работой сайта с помощью произвольно размещаемых гаджетов.

Контент содержит инструменты для управления информационными блоками, структурой сайта (т.е. папками и файлами), ресурсами медиа-библиотеки, а также настройки документооборота и управления документами.

Маркетинг содержит маркетинговые инструменты для работы магазина, проведения рассылок, баннерной рекламы и так далее.

Магазин позволяет осуществить продажу товаров и услуг с сайта.

Клиенты служит для интеграций сайта с порталами Битрикс24 и организации каналов связи с ними.

Сервисы наиболее зависим от наличия в системе тех или иных модулей. Полный набор функций позволяет управлять опросами, веб-формами, рекламой, рассылками, техподдержкой, форумми и почтовой системой.

Аналитика полностью посвящен функциям сбора, обработки и отображения статистики по сайтам. Этот раздел присутствует, только если установлен модуль Веб-аналитика.

Marketplace позволяет обновлять систему, устанавливать и обновлять партнерские модули и решения.

Настройки позволяет управлять пользователями, валютами Интернет-магазина, шаблонами, а также настраивать прочие параметры сайта и системы.

Рис. 8 - Логические блоки функционала в административной панели «1С-Битрикс: Управление сайтом» веб-представительства ИП Коренева И.В.

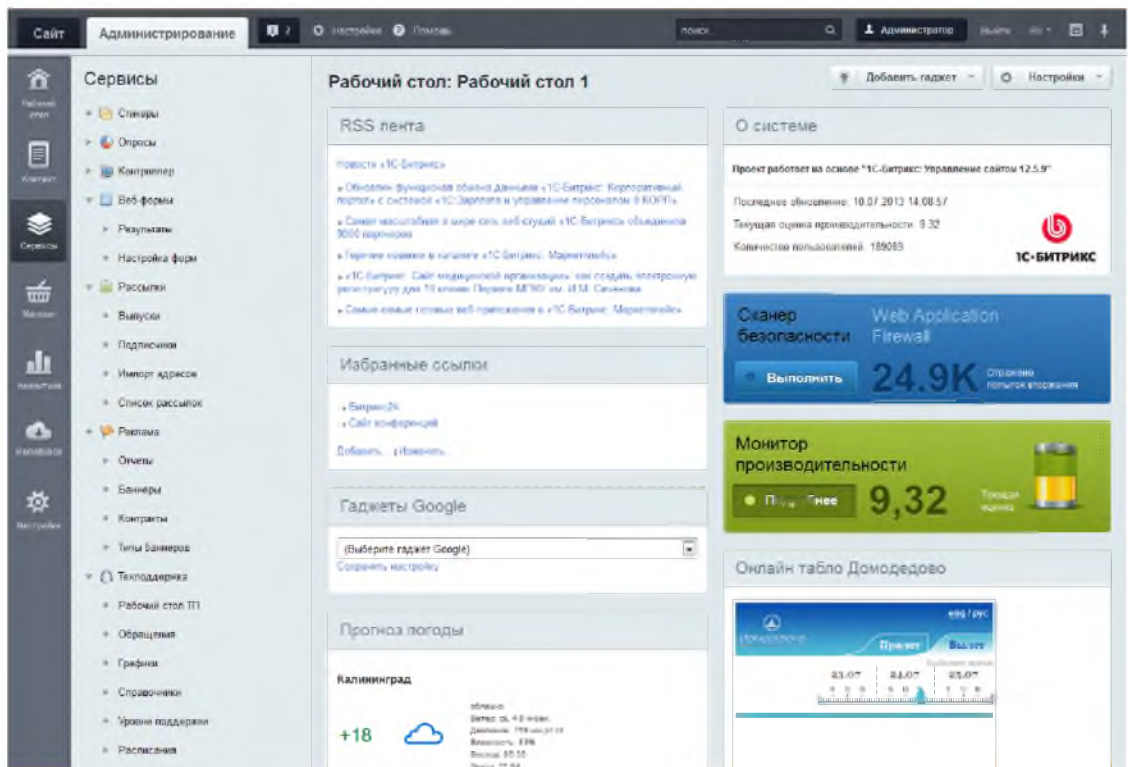


Рис. 9 - Административная панель веб-представительства

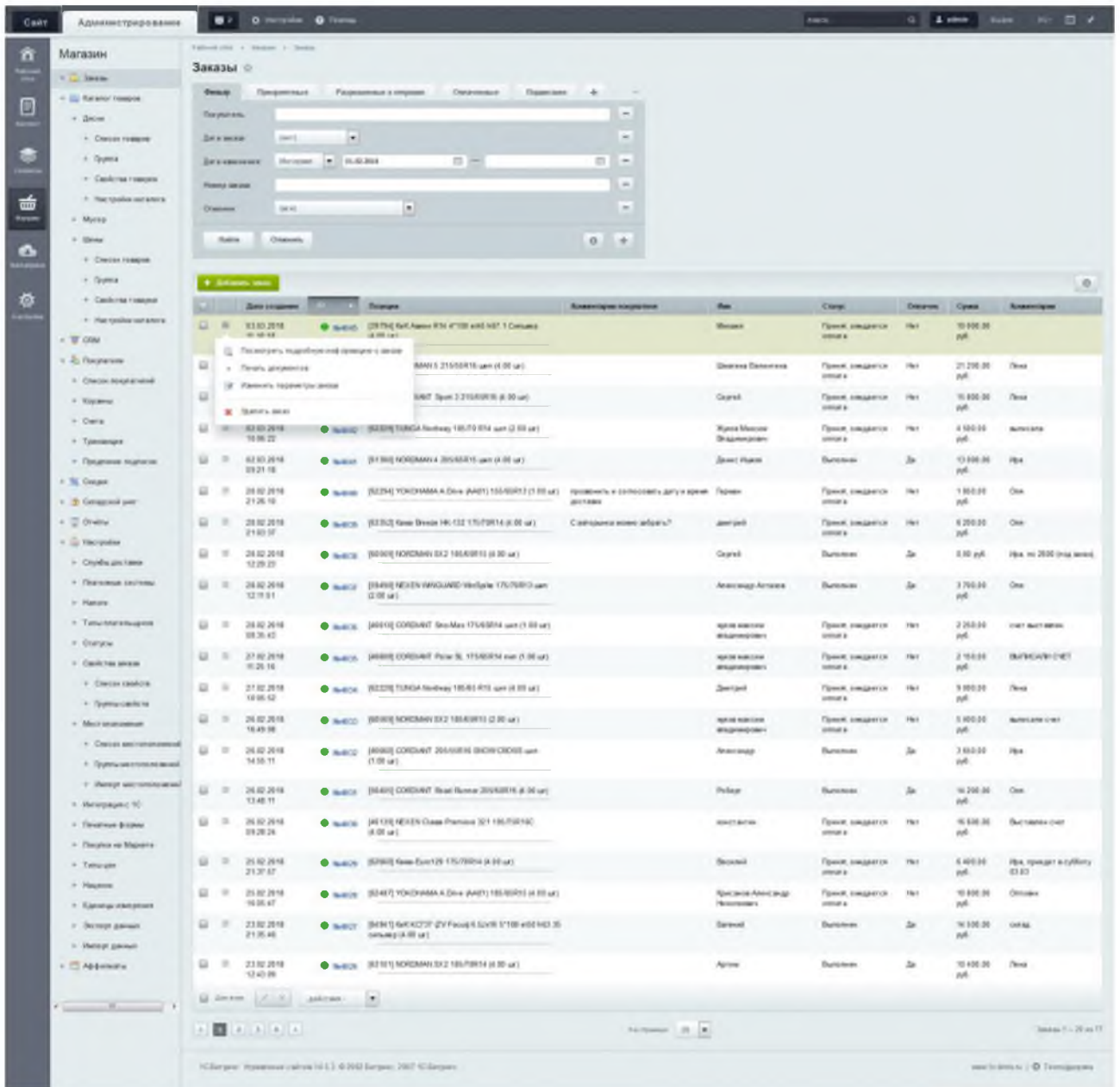


Рис. 10 - Раздел «Заказы» в Интернет-магазине

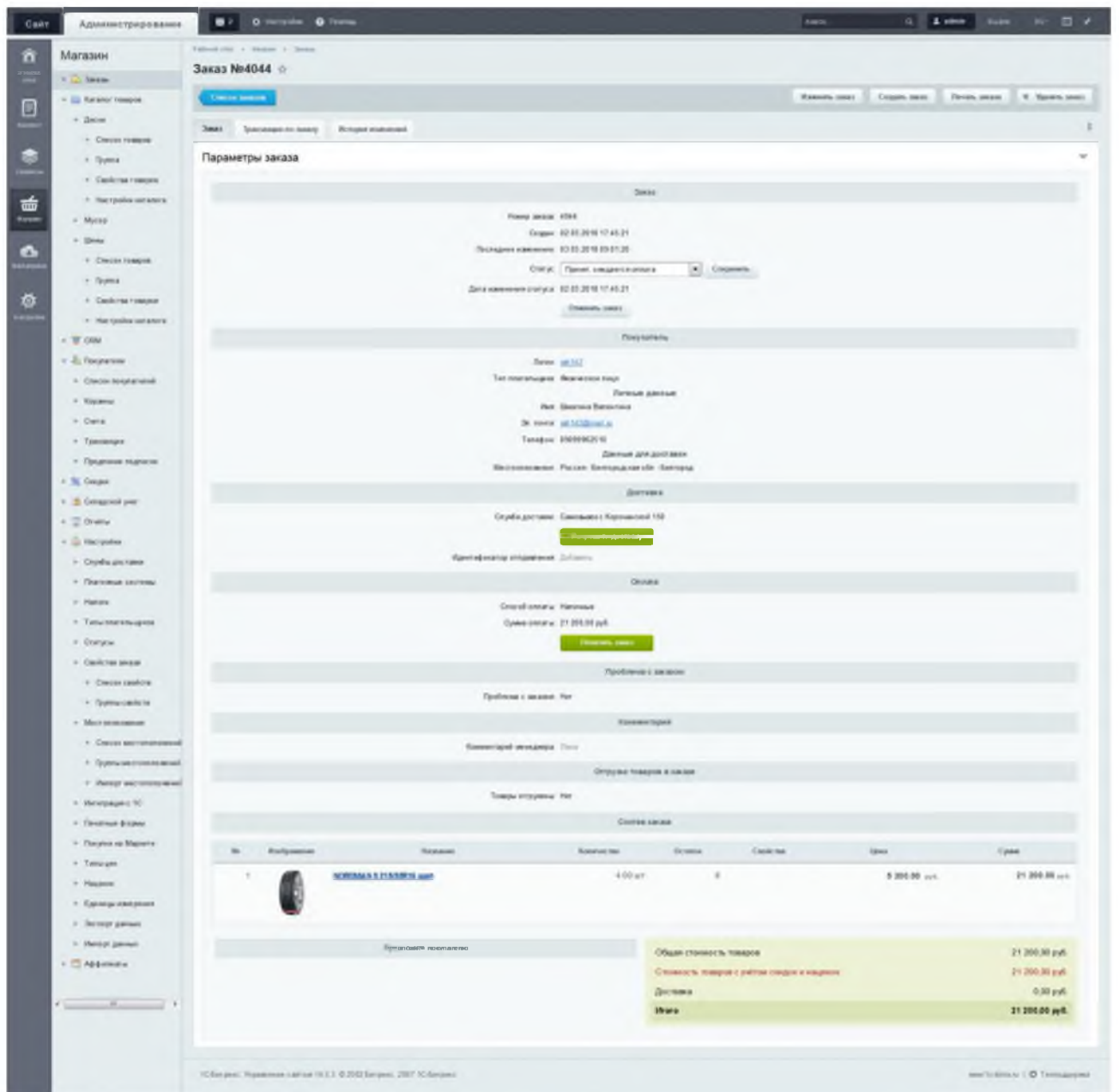


Рис. 11 - Карточка заказа клиента с номером 4044 (просмотр сведений о заказе)