МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ АВТОНОМНОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

Севастопольский государственный университет

кафедра Информационных систем

Чаусов Дмитрий Алексеевич

Институт информационных технологий и управления в технических системах

курс 4 группа ИC/б-43(о)

09.03.02 Информационные системы (уровень бакалавриата)

КУРСОВОЙ ПРОЕКТ

на тему «Интернет-Магазин цифровой техники»

по дисциплине «Веб-технологии»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Листов 65

Отметка о зачёте \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Руководитель практикума

Ст.Пр Забаштанский А.К

(должность) (подпись) (инициалы, фамилия)

Севастополь 2018

АННОТАЦИЯ

В настоящем документе описана поэтапная разработка системы. Объем работы составляет страниц – 53, количество иллюстраций –24, количество таблиц – 1, количество приложений –2, количество использованных источников –5.

Создан веб-сайт для предоставления возможности для покупки цифровой техники онлайн.

В пояснительной записке описывается выбор инструментальных средств, разработка раздела администратора, разработка пользовательской части сайта, тестирование разработанного сайта.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255807) 4

[1. ПОСТАНОВКА ЗАДАЧИ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255808) 6

[1.1 Структура сайта](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 6

[1.2 Описание страниц и основного функционала сайта](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 7

[1.3 Общие требования к разрабатываемому сайту](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 10

[2. АНАЛИЗ И ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255810) 11

[3. ПРОЕКТИРОВАНИЕ БД WEB-САЙТА](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255811) 14

[4. ВЫБОР ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255812) 18

[5. РАЗРАБОТКА РАЗДЕЛА АДМИНИСТРАТОРА](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255813) 19

[5.1 Проектирование интерфейса раздела администратора](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 19

[6. РАЗРАБОТКА РАЗДЕЛА ПОЛЬЗОВАТЕЛЯ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255816) 21

[6.1 Проектирование интерфейса раздела пользователя](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 21

[6.2 Примеры программных модулей](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255809) 25

[7. ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО САЙТА](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255819) 27

[7.1 Анализ кроссбраузерности сайта](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255820) 27

[7.2 Профилирование разработанного сайта](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255822) 28

[7.3 Тестовые примеры работы](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255822) 29

[ЗАКЛЮЧЕНИЕ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255823) 31

[СПИСОК ЛИТЕРАТУРЫ](file:///D:\Документы\4%20курс\WEB\Курсач\левачеллы\Veb_Kursovaya_Rabota__Poyasnitelnaya_Zapiska.docx#_Toc436255824) 32

ПРИЛОЖЕНИЕ А. Листин с примерами основных функций 33

ПРИЛОЖЕНИЕ Б. Схема 47

ВВЕДЕНИЕ

**Актуальность темы.** Актуальность данного проекта обусловлена тем, что передовым направлением маркетинговых, экономических и информационных сфер – является сеть интернет. Многим пользователям необходима возможность приобретения товаров и услуг через интернет. Следовательно, для поставщиков этих самых товаров, необходимо предоставить данную возможность, посредству Web-систем. Данная сфера является одной из самых прибыльных и эффективных способов рекламы и предоставления самых разнообразных услуг в данное время.

**Цель и задачи работы.** Целью данной работы является создание веб-сайта, в котором будет реализована возможность удаленного оформления заказа на цифровую технику.

В процессе выполнения настоящей работы был выполнен ряд задач:

* постановка задачи;
* анализ и описание предметной области;
* проектирование БД Веб-сайта;
* выбор инструментальных средств;
* разработка раздела администратора;
* разработка раздела пользователя;
* тестирование разработанного сайта.

**Практическое значение работы.** Веб-сайт, созданный в результате выполнения данной работы, может представлять интерес для профильной организации, а также для заинтересованных покупателей. Любой пользователь может зарегистрироваться в личном кабинете и оформить заявку на приобретение интересующей его продукции.

**Структура работы.** Настоящая работасостоит из пояснительной записки, включающей в себя введение, восемь разделов, заключение, список литературы, и приложения.

В первом разделе описывается постановка задачи, в виде пояснительной записки, описывающей основные требования к веб-сайту.

Во втором разделе проведен анализ предметной области, где описано функционирование системы с учетом конкретных знаний о предметной области.

Третий раздел посвящен проектированию и разработке БД, а также её нормализация. В этом разделе проиллюстрирована ER-диаграмма БД, а также структура БД.

Одним из главных этапов при проектировании системы является «Выбор инструментальных средств», который подробнее описан в четвертом разделе.

Пятый раздел описывает раздел сайта для администратора.

Шестой раздел описывает модераторскую часть сайта.

В седьмом разделе описывается основной раздел сайта – пользовательский.

В восьмом разделе описывается процесс тестирования функциональности данного проекта и его соответствие требованиям.

В девятом разделе, предоставлено описание основной структуры сайта, его содержание в виде файлов, а также краткое описание каждого из них.

1. ПОСТАНОВКА ЗАДАЧИ

Целью курсового проектирование является закрепление и обобщение знаний по дисциплине «Веб-технологии», приобретение и закрепление практических навыков использования современных средств и технологий в области проектирования и разработки веб-сайта.

Этапы выполнения курсового проекта:

1. Изучение предметной области, разработка и утверждение технического задания на проектирование веб-сайта.
2. Разработка дизайна веб-сайта.
3. Создание HTML макета веб-сайта.
4. Разработка БД веб-сайта.
5. Разработка раздела администратора веб-сайта.
6. Разработка пользовательской части веб-сайта.

1.1 Структура сайта

В рамках данного курсового проекта разрабатывается сайт цифрового магазина. Настоящий сайт делится на несколько разделов: раздел пользователя, раздел модератора, и раздел администратора.

Структура пользовательского раздела описана ниже:

* главная страница;
* каталог;
* контакты;
* корзина;
* личный кабинет;
* авторизация;

Структура раздела администратора:

* новости;
* список пользователей;
* каталог продукции;
* заявки на покупку.

Структура раздела модератора:

* новости;
* список пользователей.

Структура меню сайта:

* главная страница;
* каталог;
* контакты;
* Поиск товаров;
* Категории товаров.

1.2 Описание страниц и основного функционала сайта

Подключение в Index.php, ранее созданных блоков, которые показывают header, footer, боковую панель, поиск, навигационную цепочку.



Рисунок 1 – Index.php

Блок header, который отвечает за информацию, предоставленную сверху сайта, так же, если пользователь войдет на сайт, то в этом блоке сработает код приветствия и выведет сообщение “Здравствуйте, <логин>”



Рисунок 2 – block-header.php

Блок footer, отвечает за информацию, предоставленную снизу сайта, с возможностью обратной связи, рассказа друзьям и о множестве другой полезной информации.



Рисунок 3 – block-footer.php

Блок Категории товаров, он предназначен для того чтобы облегчать задачу пользователя и произвести сортировку между всеми товарами и вывести то, что хочет клиент.



Рисунок 4 – block-news.php

Страница Контактов, здесь можно оставить сообщение администратору сайта, предварительно оставив: Имя, Email и Тему сообщения.

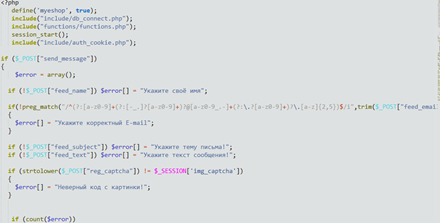




Рисунок 5,6 – feedback.php

1.3 Общие требования к разрабатываемому сайту

Разрабатываемый веб-сайт должен содержать раздел администратора, обеспечивающий возможность управления информацией, отображаемой в пользовательской части сайта. Необходимо предусмотреть авторизованный доступ к разделу администратора.

Серверная часть сайта должна разрабатываться с использованием шаблона Модель-Представление-Контроллер.

При разработке клиентской части сайта необходимым является использование одной или нескольких JavaScript библиотек.

При выполнении курсового проекта запрещается использование систем управления содержимым веб-сайтов.

2 АНАЛИЗ И ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

**Определим основные задачи, которые перед нами ставит «заказчик»:**

1. Необходимо создать многостраничный сайт с удобным, простым и интуитивно понятным интерфейсом.
2. Необходимо создать раздел новостей, которые могут быть добавлены и отредактированы модератором или администратором сайта.
3. Необходимо реализовать раздел «каталог товаров», в котором будет предоставлена двухуровневая иерархия списка товаров, в нём содержится информация о предлагаемых товарах.
4. Пользователь может выбрать отдельный вид товара и оформить заказ на его покупку. Заказ будет рассмотрен администратором, и в случае успешного подтверждения заказа, администратор или модератор свяжется с пользователем и сообщит ему дальнейшие нюансы сделки.
5. Необходимо реализовать раздел «контактов», где пользователь сможет узнать контактные данные представителей фирмы, дабы связаться с ними и задать им интересующие пользователя вопросы.
6. Необходимо реализовать Личный кабинет, через который пользователи с разными уровнями доступа (пользователь, модератор, администратор) смогут получить возможность управления ресурсами системы, в зависимости от их уровня доступа:
   1. Пользователь – оформление заявки на покупку, просмотр её состояния в личном кабинете.
   2. Модератор – просмотр списка зарегистрированных пользователей. Добавление и редактирование новостей в разделе «новости».
   3. Администратор – редактирование списка пользователей, добавление и редактирование новостей, добавление и редактирование товаров в базу данных, просмотр заявок на покупку и работа с ними.
7. Система должна быть понятной пользователю, и максимально простой.

**Основываясь на приведённых выше задачах, смоделируем процесс оформления заявки пользователем и последовательности его действий:**

1. Пользователь попадает на наш сайт.
2. Пользователь обращает внимание на информацию о наших ценах, скидках, акция в соответствующих разделах.
3. После изучения раздела новостей и раздела каталога, пользователь принимает решение об оформления заявки на покупку конкретного товара.
4. Регистрируется/авторизируется в системе.
   1. В случае, если пользователь попытается оформить заказ на покупку – он должен быть перенаправлен на страницу авторизации/регистрации.
5. Авторизованный пользователь выбирает интересующую его категорию, выбирает товары из приведённого списка. Пользователь может приобрести товар с помощью предусмотренной кнопки, или узнать подробную информацию о товаре, после чего приобрести его, или вернуться к просмотру каталога товаров.
6. Пользователь оформляет заявку на приобретение товара.
7. Заявка попадает в БД.
8. Администратор проводит мониторинг заявок на покупку, принимает решение об одобрении или отказе в заявке пользователю. В случае принятия положительного решения о сделке – администратор связывается с пользователем, по контактному номеру телефона пользователя или его эмейлу, указанных при регистрации.
9. Сделка происходит. Пользователь приобретает товар. Администратор удаляет «отработанную» заявку из базы данных.

Альтернативный вариант: Заинтересованный пользователь связывается с поставщиком посредству данных на странице «контакты» или посещает офис поставщика и совершает сделку «на месте».

**Краткое описание работы системы:**

Можно выделить три сущности, взаимодействующие с системой: пользователь, администратор и модератор.

Основная цель администратора – редактирование и контроль информации, находящейся в базе данных, которая частично используется в пользовательском интерфейсе сайта. После авторизации администратору открывается возможность удаления, редактирования его контента: информация товарах, информация о заявках, новости сайта.

Модератор имеет возможность добавлять и редактировать новости в своём личном кабинете и просматривать список пользователей.

Пользователь имеет возможность ознакомиться со всеми разделами сайта, включая Новости, Каталог товаров, информацию о конкретных товарах, получить контакты организации, узнать местонахождения офиса организации, зарегистрироваться в системе, оформить заявку на покупку конкретного товара, которая, в последствии, будет принята или отклонена администратором.

3 ПРоектирование базы данных системы

Проектирование базы данных выполняется на основании результатов анализа предметной области.

На первом этапе требуется составить логическую модель, представленную на рисунке 7 Логическая модель позволяет понять суть проектируемой системы, отражая логические взаимосвязи между сущностями.

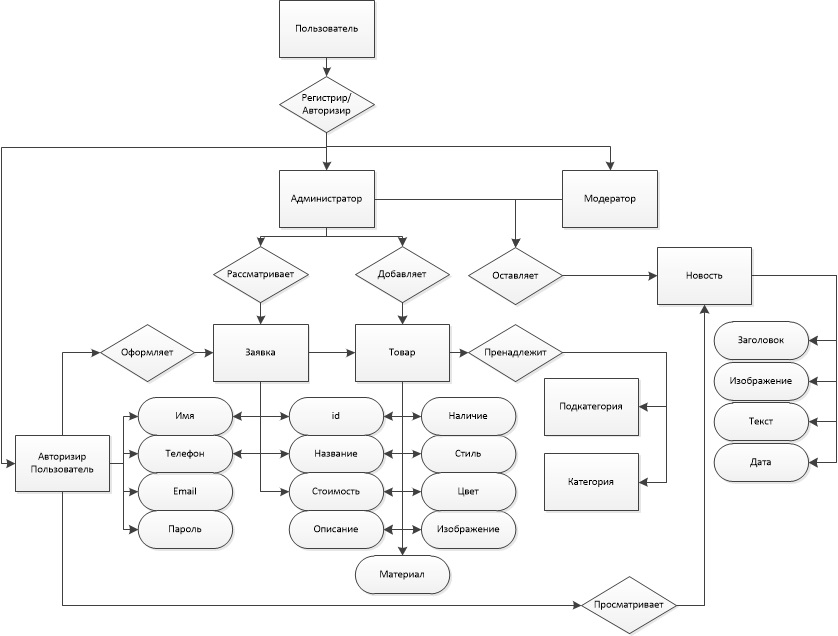


Рисунок 7 – Логическая модель

(Данная схемы предоставленна в более крупном варианте в приложении В)

Опишем сущности, приведённые в схеме выше:

1. Сущность «Пользователь»: имеет возможность зарегистрироваться, или авторизироваться, в случае, если его данные уже зарегистрированы в системе.
2. Сущность «Новость»:

Имеет атрибуты:

2.1 id.

* 1. Заголовок.
  2. Текст.
  3. Дата.

1. Сущность «Товар»:

Имеет атрибуты:

* 1. Id.
  2. Тип.
  3. Бренд.

1. Сущность «Заявка»:

Имеет атрибуты:

* 1. Имя пользователя-заявителя.
  2. Телефон пользователя-заявителя.
  3. Id приобретаемого товара.
  4. Название приобретаемого товара.
  5. Стоимость приобретаемого товара.

1. Сущность «Авторизированный пользователь»:

5.1. \*Оформляет\* «Заявку» на покупку «Товара».

5.2. \*Просматривает\* «Новости».

6. Сущность «Администратор»:

6.1. \*Рассматривает\* «Заявку» «Авторизированного пользователя» на приобретения «Товара».

6.2. \*Добавляет\* «Товар» в БД.

6.3. \*Оставляет\* «Новость».

7. Сущность «Модератор» - \*Оставляет\* «Новость».

На основании выше перечисленного можно сделать вывод, что в данном описании информационная система находится в 1 Нормальной Форме (далее НФ), так как присутствует наличие первичных ключей и атомарность атрибутов.

Приведение ко 2НФ можно опустить, так как на предыдущем этапе проверке отношений на наличие неполных функциональных зависимостей уже было определено.

Для 3НФ характерна проверка отсутствия транзитивных зависимостей, и возможном их устранений. Транзитивная зависимость подразумевает под собой то, что если для атрибутов А, В и С некоторого отношения существуют функциональные зависимости А→В, В→С, говорят, что атрибут С связан транзитивной зависимостью с атрибутом А через атрибут В (при этом атрибут А не должен функционально зависеть ни от атрибута В, ни от атрибута С).

Анализируя данную базу данных можно отметить, что таких отношений не выявлено и проектируемая база данных находится в третьей нормальной форме.

Приведение к 4НФ заключается в устранение многозначных зависимостей или зависимостей между ключевыми атрибутами. Т. к. проектируемая база данных не содержит составных ключей, то можно сказать, что многозначные зависимости полностью отсутствуют и проектируемая база данных находится в четвертой нормальной форме. Рассмотрение дальнейших нормальных форм не имеет смысла, ведь они подходят только в случае наличия многозначных зависимостей.

Далее следует построить физическую модель данных.

Физическая модель отражает физические свойства проектируемой базы данных (типы данных, размер полей, индексы). Параметры физической информационной модели зависят от выбранной системы управления базами данных (СУБД) [2].

Таким образом, проектирование баз данных – процесс создания схемы базы данных и определения необходимых ограничений целостности. Основные задачи: обеспечение хранения в базе данных всей необходимой информации, обеспечение возможности получения данных по всем необходимым запросам, сокращение избыточности и дублирования данных, обеспечение целостности базы данных.

В ходе проектирования были созданы логическая модель и физическая.

Логическая модель позволяет понять суть проектируемой системы, отражая логические взаимосвязи между сущностями. Физическая модель отражает физические свойства проектируемой базы данных.

4 ВЫБОР ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ

В качестве CSS/HTML фреймворка был выбран Bootstrap. У него есть ряд преимуществ, что делает его самым популярным среди других подобных фреймворков. Преимущества Bootstrap заключаются в следующем:

* Скорость работы – создание макетов с Bootstrap занимает меньше времени благодаря большому набору готовых к использованию элементов.
* Гибкость – добавление новых элементов не нарушает общую структуру благодаря динамически изменяющейся сетке.
* Легкая изменяемость – правка стилей достигается за счет добавления новых CSS правил, которые переопределяют существующие.
* Большое количество шаблонов – этот момент будет отдельно рассмотрен далее.
* Совместимость с браузерами – Bootstrap совместим с Mozilla Firefox, Google Chrome, Safari, Internet Explorer и Opera.

При разработке сайта использовался сервер OpenServer 5.2.2, за его простоту в настройке и использовании. Для реализации базы данных использовался PhpMySql. В качестве среды разработки выбран редактор Sublime Text 2.

1. РАЗРАБОТКА РАЗДЕЛА АДМИНИСТРАТОРА
   1. Проектирование интерфейса раздела администратора

В разделе администратора пользователю предоставляется возможность управлять ресурсами системы, то есть создавать, редактировать и удалять записи. При реализации интерфейса администратора необходимо стремиться к единообразному, простому и понятному интерфейсу, формы не должны быть слишком большими, так как это приведет к сложности в их заполнении.

Администратору должны быть предоставлены возможности управлять таким контентом, как «Каталог продукции», «Список пользователей», «Список заявок на покупку».

В разделе администратора расположено три пункта меню:

* Каталог продукции;
* Список пользователей;
* Список заявок на покупку.

Основной функционал для администратора описан ниже:

* публикация новостей;
* редактирование, удаление новости;
* редактирование, каталога продукции;
* просмотр и удаление зарегистрированных пользователей;
* просмотр и удаление заявок.

Редактирование, удаление и любое изменение каких-либо записей, происходит с помощью PhpMyAdmin. Он очень удобен в использовании и все данные пользователей и много другое, всегда все под рукой.



Рисунок 7 – Каталог продукции

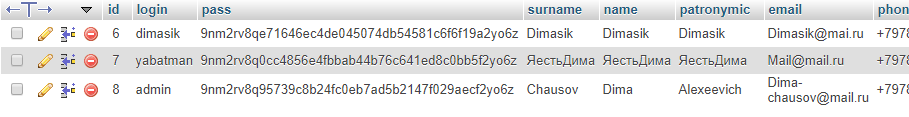


Рисунок 8 – Список пользователей



Рисунок 9 – Список заявок на покупку

Здесь представлена база данных каталога, пользователей и списка заявок на покупку продукта.

1. РАЗРАБОТКА РАЗДЕЛА ПОЛЬЗОВАТЕЛЯ
   1. Проектирование интерфейса раздела пользователя

Для полного понимания прав и выделения ролей участников строится UML диаграмма прецедентов, представленная на рисунке 10.

Данная диаграмма, отражает отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне. Выделяются актеру, которые будут взаимодействовать с данной системой и каждому актеру определятся соответствующие прецеденты.

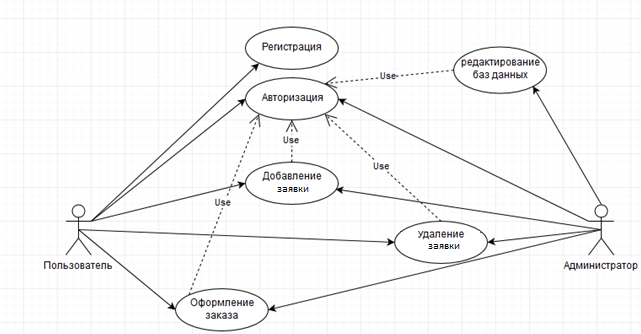


Рисунок 10 – UML диаграмма прецендентов

Исходя из диаграммы, изображенной на рисунке 10, можно представить жизненный цикл пользователя-покупателя. Схема жизненного цикла представлена на рисунке 11.

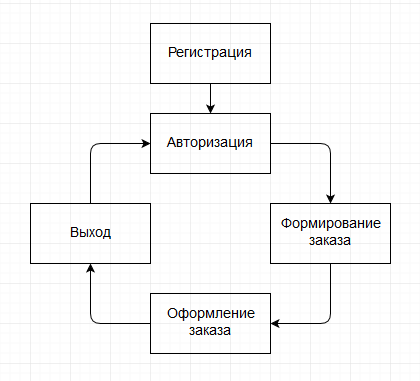


Рисунок 11 – Модель жизненного цикла пользователя

Так как в работе сайта отмечено разграничение прав доступа на две типа, то будут уместны два разных интерфейса. Пользовательский — общий для всех и административный, доступный администратору

Первым шагом для определения типа пользователя служит регистрация или вход в систему (рисунок 12).

В пользовательской части находится основной функционал сайта. Пользователь, зайдя на сайт, может авторизоваться/зарегистрироваться и ему предоставится дополнительный функционал, либо может воспользоваться частью предоставленного функционала. Без авторизации пользователь может просматривать все страницы, но ему не будет доступна возможность оценить компанию и оставить отзыв. Форма авторизации проиллюстрирована на рисунке 12.

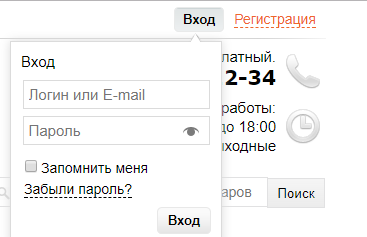


Рисунок 12 – Форма авторизации пользователя

В качестве логина пользователь может вводить Email, который полностью идентифицирует пользователя.

Страница «Каталог товаров» содержит основную информацию наличии товара, изображение, соответствующее тематике сайта. На этой странице пользователь может ознакомиться с описаниями.

Страница «Новинки» содержит список упорядоченных по дате новостей (рисунок 13)



Рисунок 13 – Список новостей.

Страница «Контакты» позволяет оставить текст сообщения.

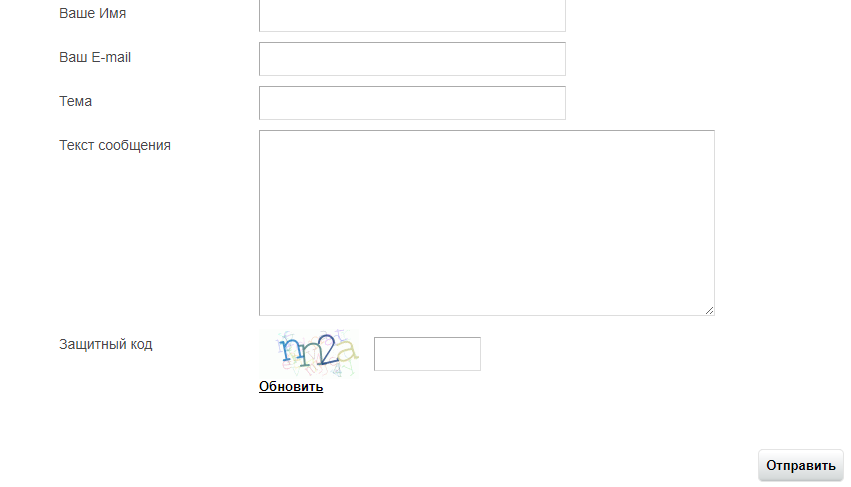


Рисунок 14 – Страница “Контакты”

Основной страничкой сайта является страница «Главная». На ней выводятся поступление товаров, упорядоченные по категориям. В каждом блоке (рисунок 15) выводится основная информация о товаре, его цена, описание, а также кнопка «Детали», при нажатии на которую пользователь переходит на страницу «Выбранного товара».

Интерфейс компании выглядит следующим образом:

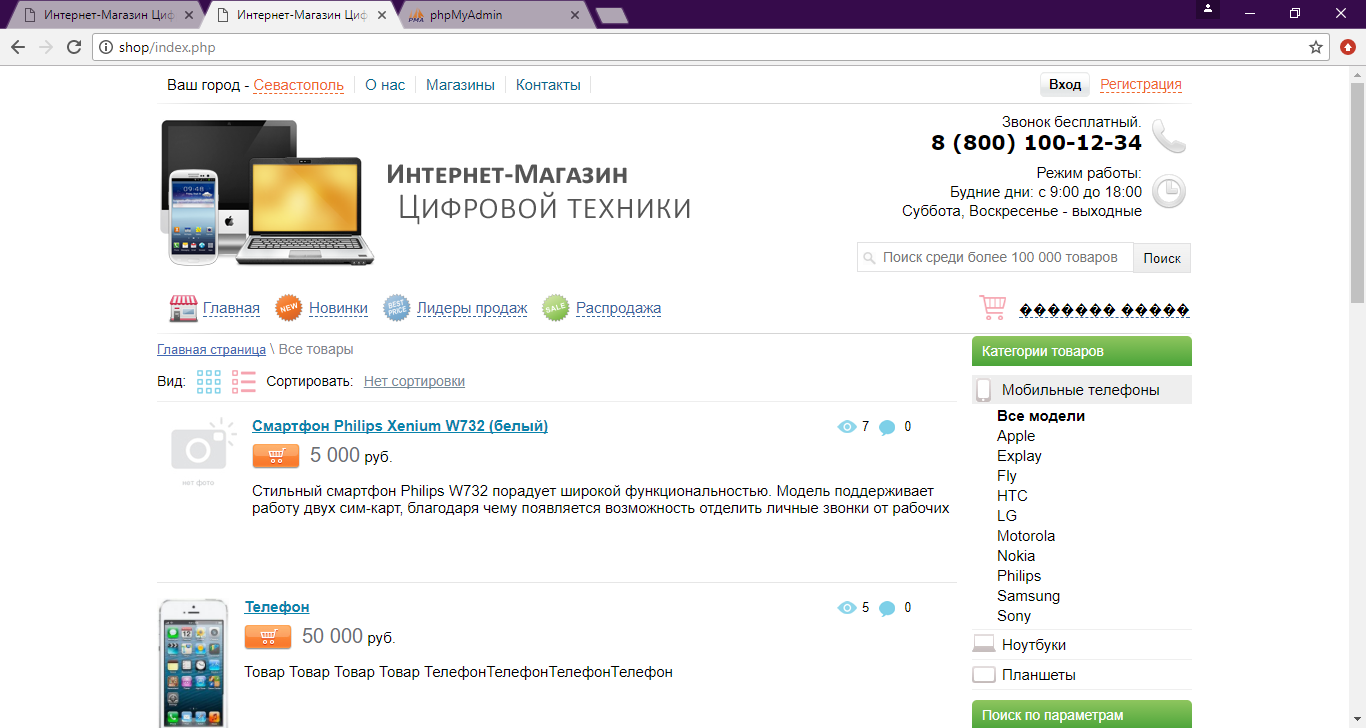


Рисунок 15 – Блок основной страницы сайта.

Список товаров показан на рисунке 16.

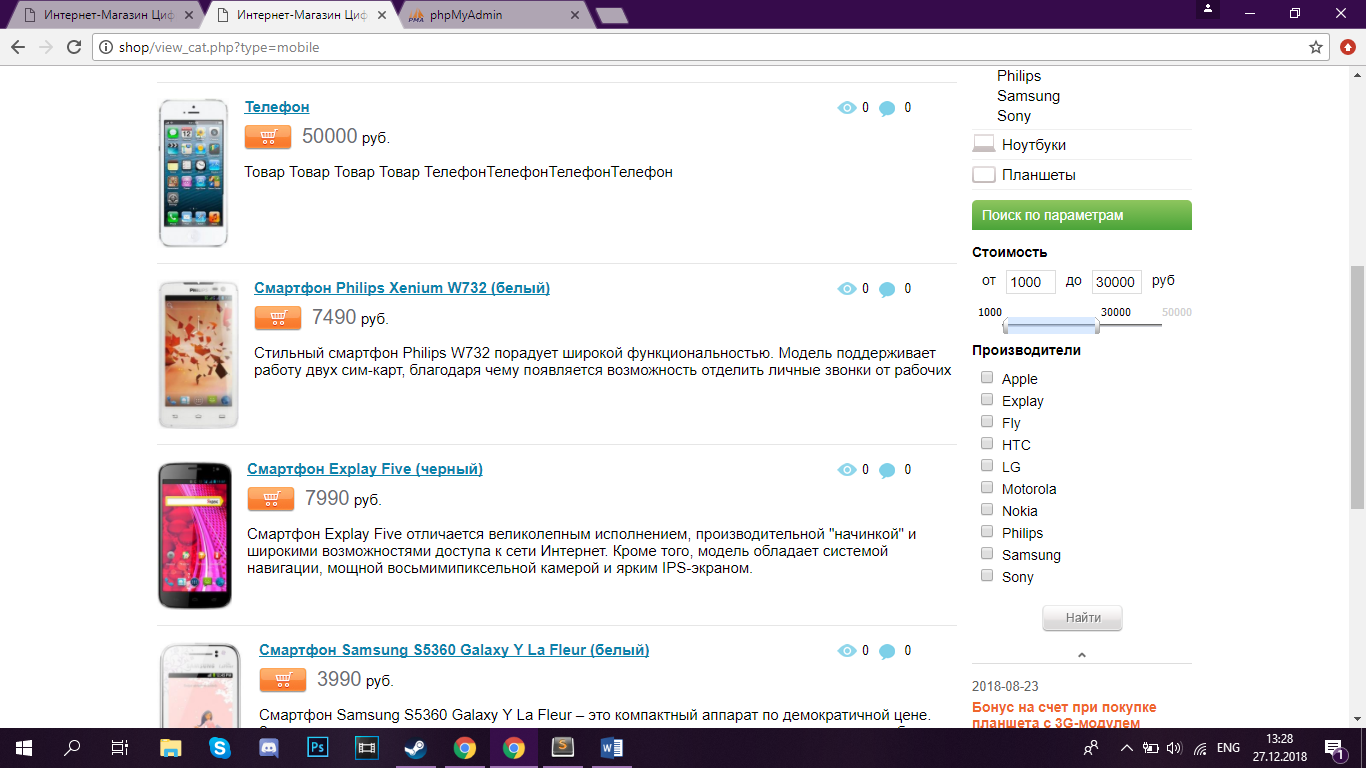


Рисунок 16 – Список товаров

6.2. Примеры программных модулей:



Рисунок 17 – add\_review.php

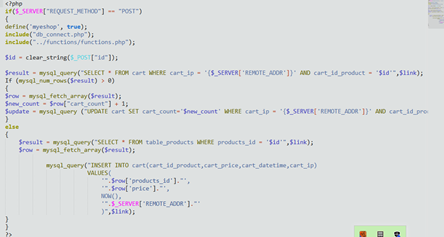


Рисунок 18 – addtocart.php



Рисунок 19 – loadcart.php

Здесь приведены функционально завершающие фрагменты программы, которые предназначены для хранения, объединения и взаимодействия с другими программными модулями.

7 ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО САЙТА

7.1 Анализ кроссбраузерности сайта

Кроссбраузерность веб-приложения была протестирована на следующих веб-браузерах: Mozilla Firefox, Google Chrome, Opera. На рисунках 18, 19, 20, показана страница сайта «Главная страница», открытая в разных браузерах.

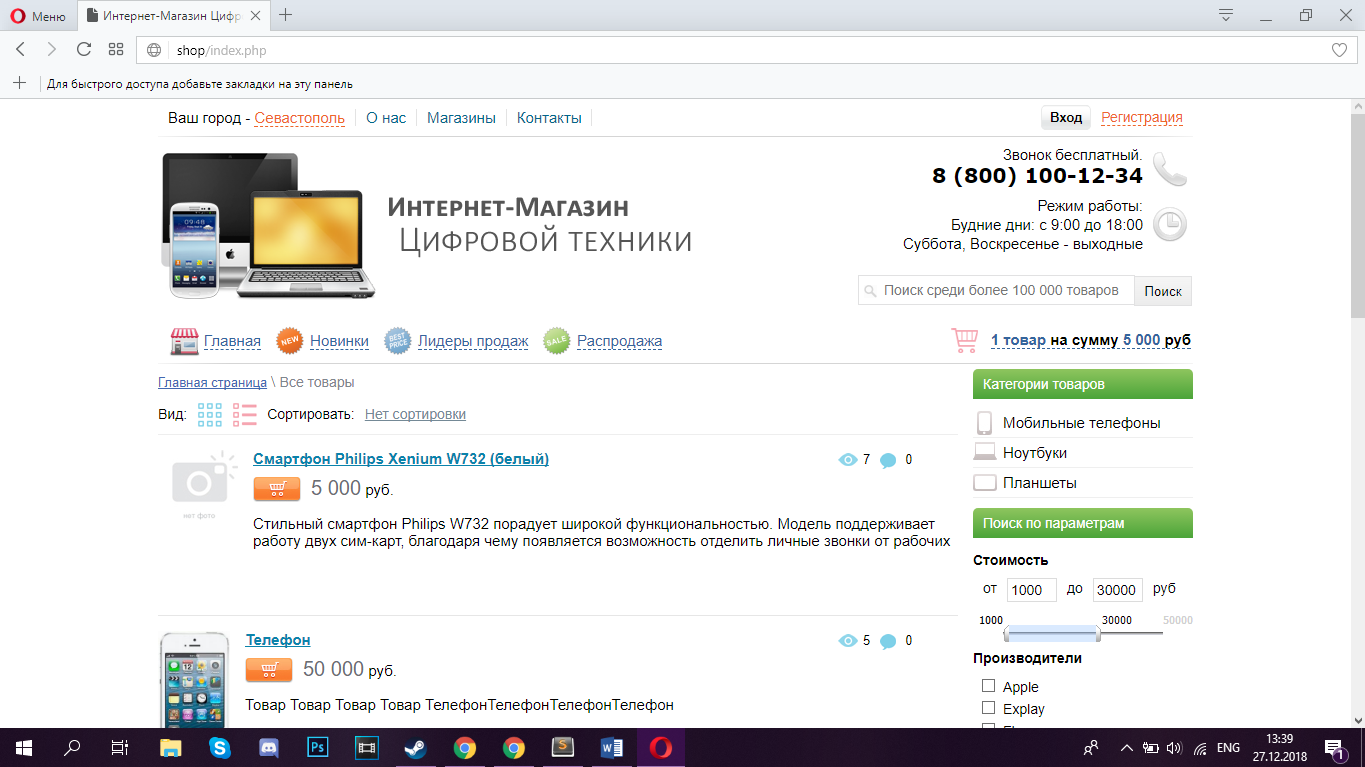


Рисунок 20 – Браузер Опера

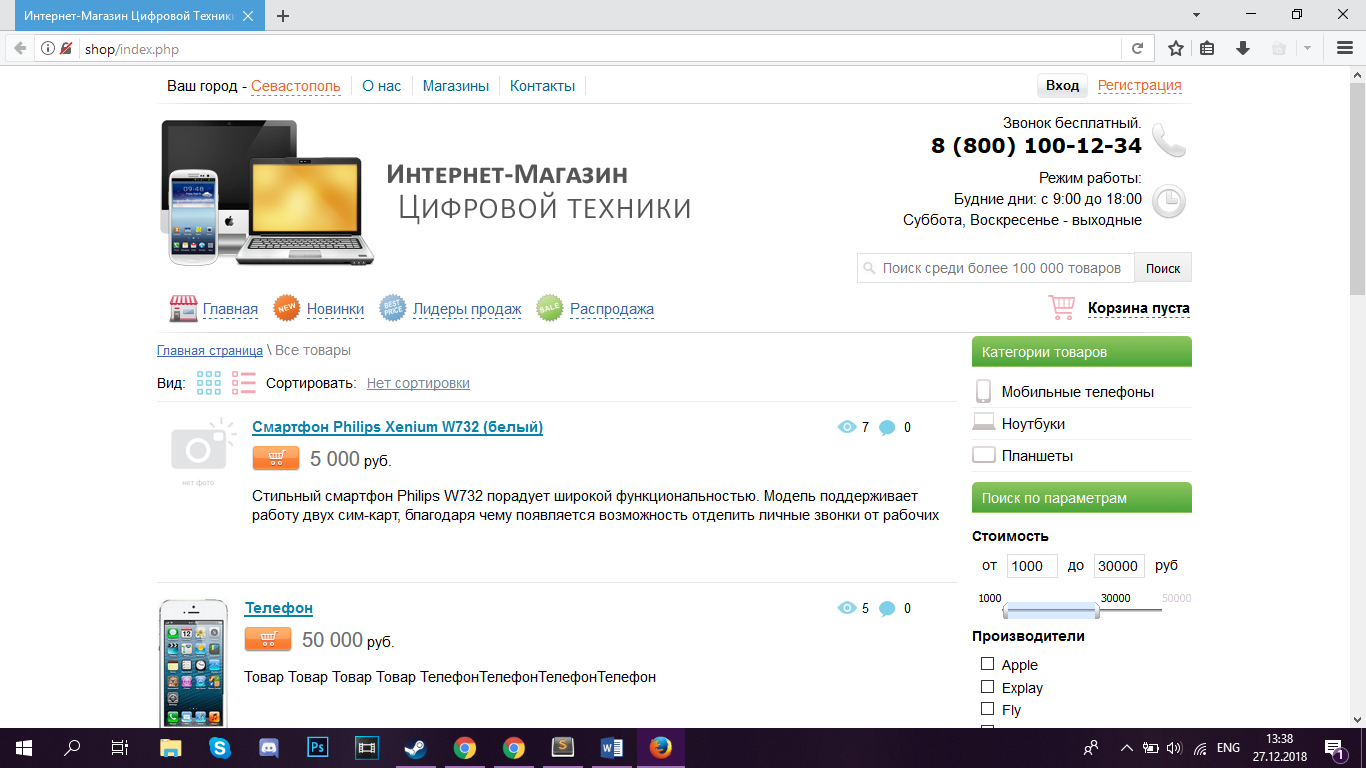


Рисунок 21 – Браузер Mozilla Firefox

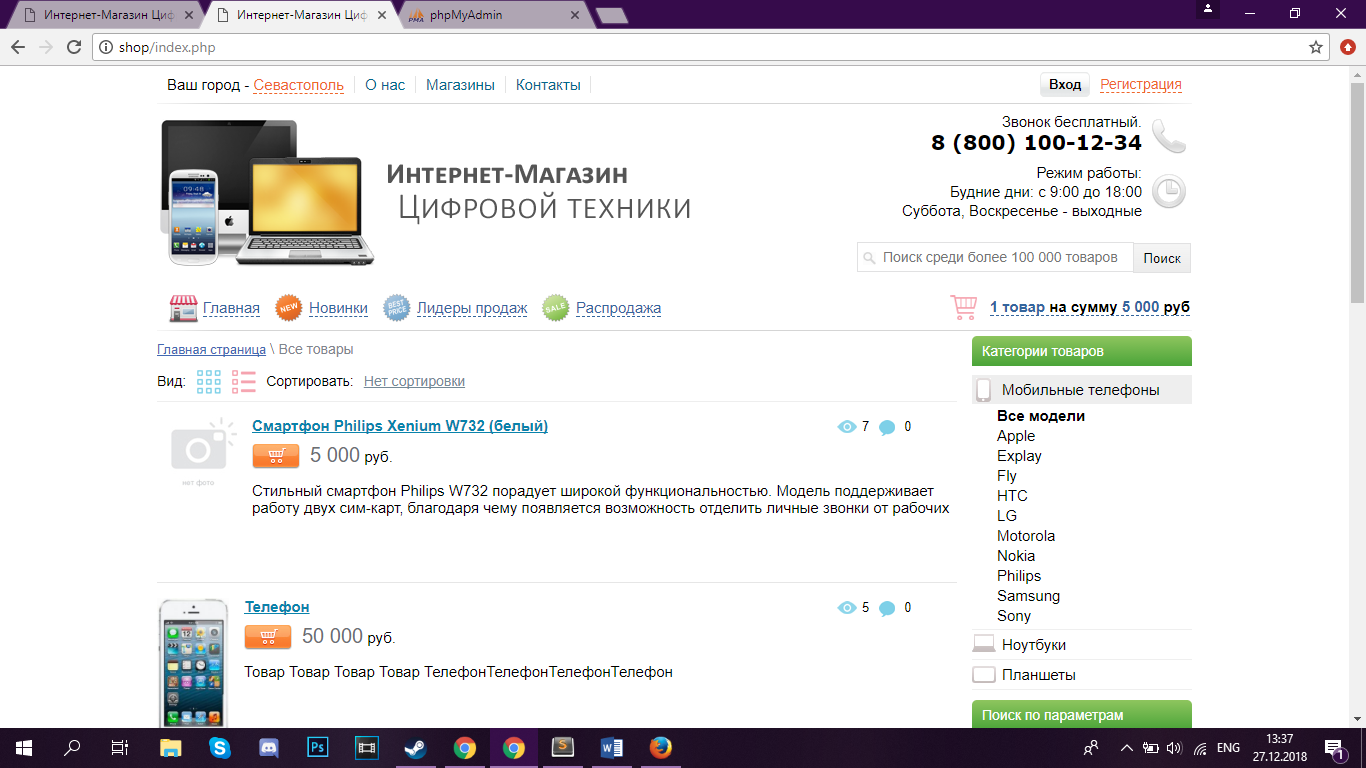


Рисунок 22 – Браузер Google Chrome

Из скриншотов видно, что сайт одинаково выглядит во всех четырех популярных браузерах. Таким образом, можно сделать вывод, что разработанное веб-приложение является кроссбраузерным.

* 1. Профилирование разработанного сайта

При выполнении профилирования разработанного сайта было определено время загрузки страниц сайта и объем передаваемого трафика, результаты представлены в таблице 7.1.

Таблица 7.1 – Время загрузки и объем переданного трафика страниц

|  |  |  |
| --- | --- | --- |
| Наименование страницы | Время загрузки, мс | Объем переданного трафика, КВ |
| Главная | 829 | 52 |
| Новости | 558 | 158 |
| Каталог | 605 | 45,7 |
| Контакты | 1401 | 58 |
| Личный кабинет | 431 | 15 |
| Авторизация | 708 | 378 |

Из содержания таблицы 7.1 видно, что сайт загружается довольно быстро, на некоторых страницах время загрузки на порядок выше среднего времени, это связано с тем, что эти страницы динамические и на общее время влияет выборка из БД.

* 1. Тестовые примеры работы сайта

В данном разделе будут продемонстрированы страницы сайта, а также основная его функциональность: поиск компании, добавление отзыва компании.

Вид главной страницы сайта можно увидеть на рисунке 21.

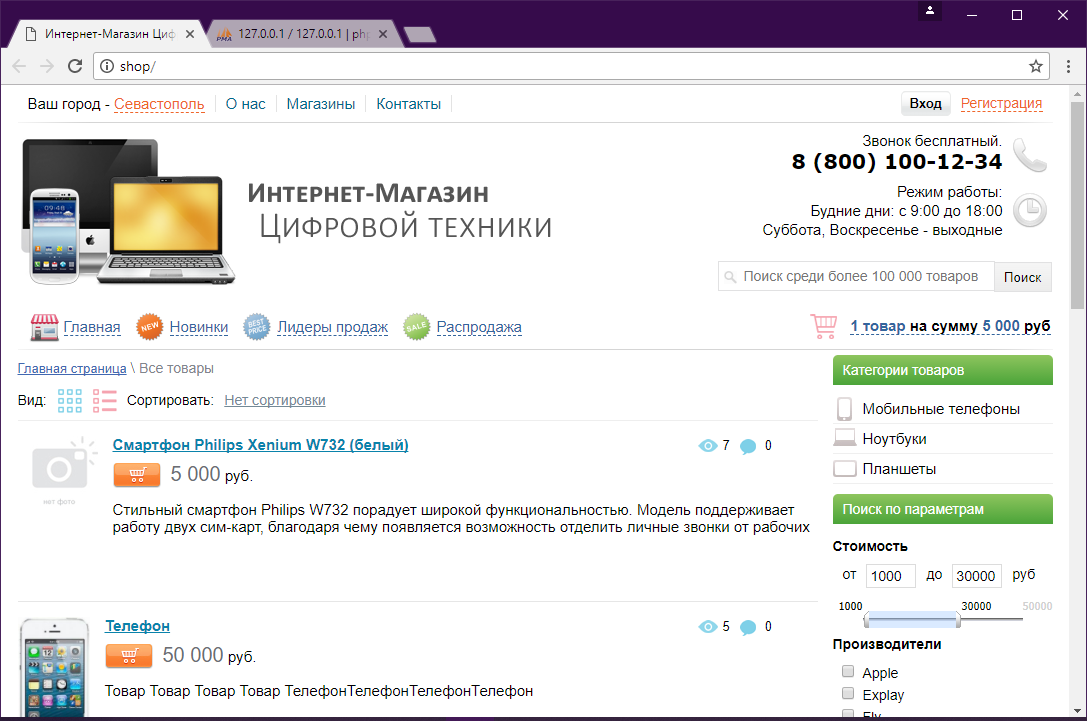


Рисунок 23 – Главная страница сайта

На этой странице расположены все товары, с возможностью сортировки. Справа расположены категории и поиск по параметрам, для облегченного поиска товаров. Имеется и поисковик товаров, так же для упрощения поиска нужного товара.

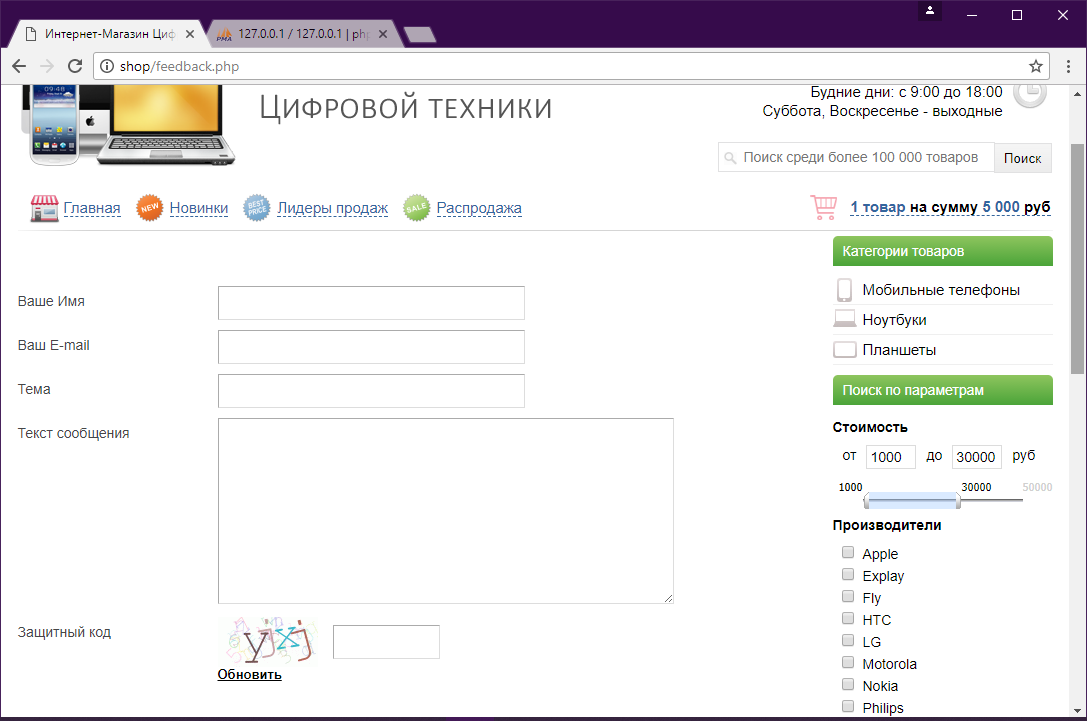


Рисунок 24 – Страница «Контакты»

Нажав на вкладку контактов, пользователь может оставить свое сообщение, пожелание или найденный баг.

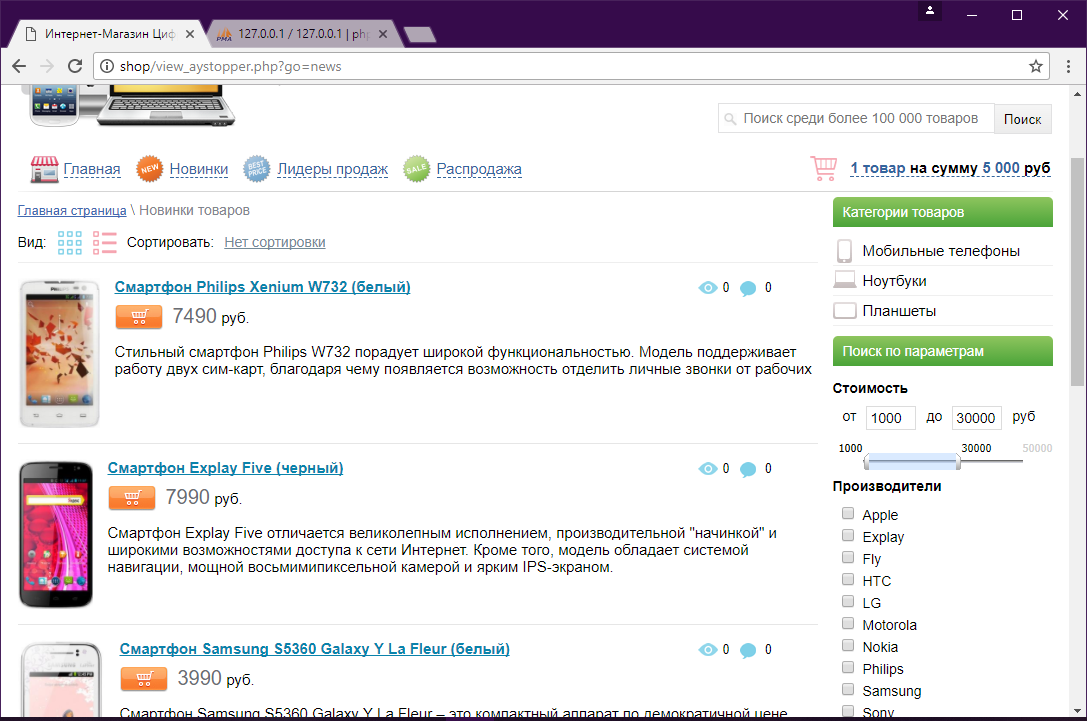


Рисунок 25 – Вкладка «Новинки»

На рисунке 25, предоставлена информация о всех новых поступлениях в продажу и если пользователь захочет посмотреть, новые поступления в продажу, то сразу будет осведомлен.

ЗАКЛЮЧЕНИЕ

В настоящем курсовом проекте был реализован веб-сайт производителя цифровой техники. Для достижения поставленной цели был выполнен анализ предметной области и поэтапные проектирование и разработка системы.

Согласно поставленной задаче, в настоящем проекте был выполнен ряд задач:

1. Проектирование и нормализация БД, третий раздел документа. База данных приведена ко второй нормальной форме, чего достаточно для уровня настоящего проекта. Если при расширяемости потребуется приведение БД к высшим нормальным формам – будет произведен рефакторинг БД и программного кода, что не составит большого объема работы.
2. Валидация полей ввода реализована при авторизации пользователя.
3. Использование JavaScript библиотеки – Jquery.
4. Верстка страниц с учетом кроссбраузерности сайта. В качестве инструмента выступил фреймворк Bootstrap, с помощью которого скорость и качество верстки значительно повысились.
5. Профилирование сайта, с проверкой скорости загрузки сайта и объема передаваемой информации.

Проектирование и правильный выбор инструментальных средств положительно повлияли на разработку веб-сайта. С учетом перечисленного списка выполненных задач, можно считать, что цели курсового проектирования были достигнуты.

В ходе выполнения курсового проектирования были закреплены навыки создания web-сайтов с использованием шаблона проектирования модель-представление-контроллер и разработан веб-сайт цифрового магазина.

СПИСОК ЛИТЕРАТУРЫ

1. Квинт, И. HTML и CSS на 100 % [Текст] / И. Квинт. – М. и др. : Питер, 2008. – 352 с. (рус.).
2. Odersky M. Programming in Scala: A Comprehensive Step-by-Step Guide, 2nd Edition / M. Odersky. – artima, 2011. – 837 c. (англ.).
3. Spurlock J. Bootstrap / J. Spurlock. – Sebastopol: O’Reilly Media, 2013. –

110 с.

1. Колисниченко, Д. Н. PHP 5/6 и MySQL 6: разработка Web-приложений [Текст] / Д. Н. Колисниченко. – 2-е изд. – СПб. : БХВ – Петербург, 2010. – 546 с. (рус.).
2. Bootstrap, официальная документация. [Электронный ресурс]. URL: http://bootstrap-3.ru/index.php (дата обращения: 02.12.2016).

**ПРИЛОЖЕНИЕ А**

**Index.php**

<?php

define('myeshop', true);

include("include/db\_connect.php");

include("functions/functions.php");

session\_start();

include("include/auth\_cookie.php");

$sorting = $\_GET["sort"];

switch ($sorting)

{

case 'price-asc';

$sorting = 'price ASC';

$sort\_name = 'От дешевых к дорогим';

break;

case 'price-desc';

$sorting = 'price DESC';

$sort\_name = 'От дорогих к дешевым';

break;

case 'popular';

$sorting = 'count DESC';

$sort\_name = 'Популярное';

break;

case 'news';

$sorting = 'datetime DESC';

$sort\_name = 'Новинки';

break;

case 'brand';

$sorting = 'brand';

$sort\_name = 'Новинки';

break;

default:

$sorting = 'products\_id DESC';

$sort\_name = 'Нет сортировки';

break;

}

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<meta http-equiv="content-type" content="text/html; charset=windows-1251" />

<link href="css/reset.css" rel="stylesheet" type="text/css" />

<link href="css/style.css" rel="stylesheet" type="text/css" />

<link href="trackbar/trackbar.css" rel="stylesheet" type="text/css" />

<script type="text/javascript" src="/js/jquery-1.8.2.min.js"></script>

<script type="text/javascript" src="/js/jcarousellite\_1.0.1.js"></script>

<script type="text/javascript" src="/js/shop-script.js"></script>

<script type="text/javascript" src="/js/jquery.cookie.min.js"></script>

<script type="text/javascript" src="/trackbar/jquery.trackbar.js"></script>

<script type="text/javascript" src="/js/TextChange.js"></script>

<title>Интернет-Магазин Цифровой Техники</title>

</head>

<body>

<div id="block-body">

<?php

include("include/block-header.php");

?>

<div id="block-right">

<?php

include("include/block-category.php");

include("include/block-parameter.php");

include("include/block-news.php");

?>

</div>

<div id="block-content">

<div id="block-sorting">

<p id="nav-breadcrumbs"><a href="index.php" >Главная страница</a> \ <span>Все товары</span></p>

<ul id="options-list">

<li>Вид: </li>

<li><img id="style-grid" src="/images/icon-grid.png" /></li>

<li><img id="style-list" src="/images/icon-list.png" /></li>

<li>Сортировать:</li>

<li><a id="select-sort"><?php echo $sort\_name; ?></a>

<ul id="sorting-list">

<li><a href="index.php?sort=price-asc" >От дешевых к дорогим</a></li>

<li><a href="index.php?sort=price-desc" >От дорогих к дешевым</a></li>

<li><a href="index.php?sort=popular" >Популярное</a></li>

<li><a href="index.php?sort=news" >Новинки</a></li>

<li><a href="index.php?sort=brand" >От А до Я</a></li>

</ul>

</li>

</ul>

</div>

<ul id="block-tovar-grid" >

<?php

$num = 6; // Здесь указываем сколько хотим выводить товаров.

$page = (int)$\_GET['page'];

$count = mysql\_query("SELECT COUNT(\*) FROM table\_products WHERE visible = '1'",$link);

$temp = mysql\_fetch\_array($count);

If ($temp[0] > 0)

{

$tempcount = $temp[0];

// Находим общее число страниц

$total = (($tempcount - 1) / $num) + 1;

$total = intval($total);

$page = intval($page);

if(empty($page) or $page < 0) $page = 1;

if($page > $total) $page = $total;

// Вычисляем начиная с какого номера

// следует выводить товары

$start = $page \* $num - $num;

$qury\_start\_num = " LIMIT $start, $num";

}

$result = mysql\_query("SELECT \* FROM table\_products WHERE visible='1' ORDER BY $sorting $qury\_start\_num ",$link);

if (mysql\_num\_rows($result) > 0)

{

$row = mysql\_fetch\_array($result);

do

{

if ($row["image"] != "" && file\_exists("./uploads\_images/".$row["image"]))

{

$img\_path = './uploads\_images/'.$row["image"];

$max\_width = 200;

$max\_height = 200;

list($width, $height) = getimagesize($img\_path);

$ratioh = $max\_height/$height;

$ratiow = $max\_width/$width;

$ratio = min($ratioh, $ratiow);

$width = intval($ratio\*$width);

$height = intval($ratio\*$height);

}else

{

$img\_path = "/images/no-image.png";

$width = 110;

$height = 200;

}

// Количество отзывов

$query\_reviews = mysql\_query("SELECT \* FROM table\_reviews WHERE products\_id = '{$row["products\_id"]}' AND moderat='1'",$link);

$count\_reviews = mysql\_num\_rows($query\_reviews);

echo '

<li>

<div class="block-images-grid" >

<img src="'.$img\_path.'" width="'.$width.'" height="'.$height.'" />

</div>

<p class="style-title-grid" ><a href="http://shop//mobile/'.$row["products\_id"].'-'.ftranslite($row["title"]).'/" >'.$row["title"].'</a></p>

<ul class="reviews-and-counts-grid">

<li><img src="/images/eye-icon.png" /><p>'.$row["count"].'</p></li>

<li><img src="/images/comment-icon.png" /><p>'.$count\_reviews.'</p></li>

</ul>

<a class="add-cart-style-grid" tid="'.$row["products\_id"].'" ></a>

<p class="style-price-grid" ><strong>'.group\_numerals($row["price"]).'</strong> руб.</p>

<div class="mini-features" >

'.$row["mini\_features"].'

</div>

</li>

';

}

while ($row = mysql\_fetch\_array($result));

}

?>

</ul>

<ul id="block-tovar-list" >

<?php

$result = mysql\_query("SELECT \* FROM table\_products WHERE visible='1' ORDER BY $sorting $qury\_start\_num",$link);

if (mysql\_num\_rows($result) > 0)

{

$row = mysql\_fetch\_array($result);

do

{

if ($row["image"] != "" && file\_exists("./uploads\_images/".$row["image"]))

{

$img\_path = './uploads\_images/'.$row["image"];

$max\_width = 150;

$max\_height = 150;

list($width, $height) = getimagesize($img\_path);

$ratioh = $max\_height/$height;

$ratiow = $max\_width/$width;

$ratio = min($ratioh, $ratiow);

$width = intval($ratio\*$width);

$height = intval($ratio\*$height);

}else

{

$img\_path = "/images/noimages80x70.png";

$width = 80;

$height = 70;

}

// Количество отзывов

$query\_reviews = mysql\_query("SELECT \* FROM table\_reviews WHERE products\_id = '{$row["products\_id"]}' AND moderat='1'",$link);

$count\_reviews = mysql\_num\_rows($query\_reviews);

echo '

<li>

<div class="block-images-list" >

<img src="'.$img\_path.'" width="'.$width.'" height="'.$height.'" />

</div>

<ul class="reviews-and-counts-list">

<li><img src="/images/eye-icon.png" /><p>'.$row["count"].'</p></li>

<li><img src="/images/comment-icon.png" /><p>'.$count\_reviews.'</p></li>

</ul>

<p class="style-title-list" ><a href="view\_content.php?id='.$row["products\_id"].'" >'.$row["title"].'</a></p>

<a class="add-cart-style-list" tid="'.$row["products\_id"].'" ></a>

<p class="style-price-list" ><strong>'.group\_numerals($row["price"]).'</strong> руб.</p>

<div class="style-text-list" >

'.$row["mini\_description"].'

</div>

</li>

';

}

while ($row = mysql\_fetch\_array($result));

}

echo '</ul>';

if ($page != 1){ $pstr\_prev = '<li><a class="pstr-prev" href="index.php?page='.($page - 1).'">&lt;</a></li>';}

if ($page != $total) $pstr\_next = '<li><a class="pstr-next" href="index.php?page='.($page + 1).'">&gt;</a></li>';

// Формируем ссылки со страницами

if($page - 5 > 0) $page5left = '<li><a href="index.php?page='.($page - 5).'">'.($page - 5).'</a></li>';

if($page - 4 > 0) $page4left = '<li><a href="index.php?page='.($page - 4).'">'.($page - 4).'</a></li>';

if($page - 3 > 0) $page3left = '<li><a href="index.php?page='.($page - 3).'">'.($page - 3).'</a></li>';

if($page - 2 > 0) $page2left = '<li><a href="index.php?page='.($page - 2).'">'.($page - 2).'</a></li>';

if($page - 1 > 0) $page1left = '<li><a href="index.php?page='.($page - 1).'">'.($page - 1).'</a></li>';

if($page + 5 <= $total) $page5right = '<li><a href="index.php?page='.($page + 5).'">'.($page + 5).'</a></li>';

if($page + 4 <= $total) $page4right = '<li><a href="index.php?page='.($page + 4).'">'.($page + 4).'</a></li>';

if($page + 3 <= $total) $page3right = '<li><a href="index.php?page='.($page + 3).'">'.($page + 3).'</a></li>';

if($page + 2 <= $total) $page2right = '<li><a href="index.php?page='.($page + 2).'">'.($page + 2).'</a></li>';

if($page + 1 <= $total) $page1right = '<li><a href="index.php?page='.($page + 1).'">'.($page + 1).'</a></li>';

if ($page+5 < $total)

{

$strtotal = '<li><p class="nav-point">...</p></li><li><a href="index.php?page='.$total.'">'.$total.'</a></li>';

}else

{

$strtotal = "";

}

if ($total > 1)

{

echo '

<div class="pstrnav">

<ul>

';

echo $pstr\_prev.$page5left.$page4left.$page3left.$page2left.$page1left."<li><a class='pstr-active' href='index.php?page=".$page."'>".$page."</a></li>".$page1right.$page2right.$page3right.$page4right.$page5right.$strtotal.$pstr\_next;

echo '

</ul>

</div>

';

}

?

</div>

<?php

include("include/block-random.php");

include("include/block-footer.php");

?>

</div>

</body>

</html>

**Style.css**

#block-body{

width: 1035px;

height: auto;

margin: 0 auto;

}

#block-header{

width: 1035px;

height: 220px;

}

#block-right{

width: 220px;

height: auto;

float: right;

}

#block-content{

width: 800px;

height: auto;

}

#header-top-block{

width: 1035px;

height: 35px;

}

#block-footer{

width: 1033px;

height: 160px;

margin-top: 10px;

clear: both;

}

#header-top-menu li{

font: 15px sans-serif;

float: left;

margin-top: 10px;

padding-left: 10px;

padding-right: 10px;

border-right: 1px solid #E3E3E3;

}

#header-top-menu a{

font: 15px sans-serif;

color: #00678C;

text-decoration: none;

}

#header-top-menu a:hover{

color: #EF662B;

border-bottom: 1px dashed #EF662B;

}

#header-top-menu span{

color: #EF662B;

border-bottom: 1px dashed #EF662B;

}

#reg-auth-title{

margin-top: 10px;

margin-right: 10px;

float: right;

}

#reg-auth-title > a#active-button{

background: url(/images/auth-butt-bg-click.png);

}

#reg-auth-title a.top-auth{

font:bold 13px sans-serif;

text-decoration: none;

background: url(/images/auth-butt-bg.png);

padding-left: 8px;

padding-right: 8px;

padding-top: 4px;

padding-bottom: 4px;

border: 1px solid #E9EAEA;

color: black;

border-radius:5px ; /\* Остальные \*/

-webkit-border-radius:5px ; /\* safari \*/

-moz-border-radius:5px ; /\* firefox \*/

cursor: pointer;

}

#reg-auth-title a.top-auth:hover{

background: url(/images/auth-butt-bg-move.png);

}

#reg-auth-title a{

font:14px sans-serif;

text-decoration: none;

color: #EF662B;

margin-left: 10px;

border-bottom: 1px dashed #EF662B;

}

#reg-auth-title a:hover{

border-bottom:none;

}

#top-line{

background: url(/images/line.png);

width: 1035px;

height: 5px;

}

#img-logo{

margin-top: 12px;

position: absolute;

}

#personal-info{

width: 300px;

margin-left: 730px;

margin-top: 7px;

}

#personal-info p{

font: 15px sans-serif;

margin-top: 2px;

margin-bottom: 0px;

margin-right: 45px;

}

#personal-info h3{

font:bold 20px verdana;

margin-top: 0px;

margin-bottom: 10px;

margin-right: 45px;

}

#personal-info img{

margin-left: 265px;

margin-top: -45px;

position: absolute;

}

#block-search{

width: 345px;

height: auto;

margin-left: 700px;

margin-top: 23px;

z-index: 999;

position: absolute;

}

#block-search span{

width: 13px;

height: 13px;

background: url(/images/icon-search.png);

position: absolute;

margin-top: 10px;

margin-left: 6px;

}

#block-search input#input-search{

width: 250px;

height: 26px;

border: 1px solid #E1E1E2;

font: 14px sans-serif;

padding-left: 25px;

}

#button-search{

width: 58px;

height: 30px;

border: 1px solid #E1E1E2;

background: url(/images/grd-button-search-norm.png);

margin-left: -5px;

cursor: pointer;

}

#button-search:hover{

background: url(/images/grd-button-search-hover.png);

}

#top-menu{

width: 1033px;

height: 45px;

margin-top: 5px;

}

#top-menu li{

float: left;

margin-left: 10px;

margin-top: 8px;

}

#top-menu a{

font: 15px sans-serif;

color: #34659D;

margin-left: 4px;

margin-right: 3px;

text-decoration: none;

border-bottom: 1px dashed #34659D;

}

#top-menu a:hover{

color:#EF662B;

border-bottom: 1px dashed #EF662B;

}

#top-menu img{

float: left;

margin-top: -7px;

}

#block-basket{

margin-top: 0px;

padding-top: 7px;

margin-right: -3px;

}

#block-basket span{

color: #34659D;

}

#block-basket a{

font: bold 15px sans-serif;

text-decoration: none;

margin-left: 40px;

border-bottom: 1px dashed #34659D;

color: black;

}

#block-basket a:hover{

border-bottom: none;

color: black;

}

#block-basket img{

position: absolute;

margin-top: -3px;

}

#nav-line{

background: url(/images/line.png);

width: 1035px;

height: 5px;

margin-top: -3px;

}

#block-category{

width: 220px;

height: auto;

}

#block-category > ul{

margin-top: -5px;

}

.header-title{

width: 210px;

font: 14px sans-serif;

margin-top: 0px;

padding-left: 10px;

padding-top: 7px;

padding-bottom: 7px;

color: white;

background: url(/images/grd-title.png);

border-radius: 5px 5px 0 0;

-webkit-border-radius: 5px 5px 0 0;

-moz-border-radius: 5px 5px 0 0;

}

#block-category > ul > li{

border-bottom: 1px solid #EFEFEF;

}

#block-category > ul > li > a{

font: 15px sans-serif;

padding-top: 6px;

padding-bottom: 6px;

padding-left: 30px;

display: block;

cursor: pointer;

}

#block-category > ul > li > a:hover{

background-color: #EEEEEE;

}

#block-category > ul > li > a.active{

background-color: #EEEEEE;

}

#mobile-images{

position: absolute;

margin-left: -26px;

margin-top: -3px;

}

#book-images{

position: absolute;

margin-left: -30px;

margin-top: -2px;

}

**JTabs.js**

(function($){

$.fn.jTabs = function(options) {

var defaults = {

content: "div.content",

equal\_height: false,

cookies: false,

animate: false,

effect: "fade",

speed: 400

};

var options = $.extend(defaults, options);

return this.each(function() {

// object is the selected pagination element list

obj = $(this);

var objTabs = $(options.content);

var number\_of\_items = obj.children("li").size();

var tabIndex = [];

var tabs = [];

// create array of tab index items

for (i=1;i<=number\_of\_items;i++) { tabIndex[i] = obj.find("li:nth-child("+i+")"); tabIndex[i].attr("title", i); }

// create array tabs

for (i=1;i<=number\_of\_items;i++) { tabs[i] = $(options.content + "> div:nth-child("+i+")"); }

// if equal height on

if(options.equal\_height) {

var maxHeight = 0;

$(options.content).children("div").each(function(){

if ($(this).outerHeight() > maxHeight) { maxHeight = $(this).outerHeight(); }

});

$(options.content).height(maxHeight);

}

// initiate the current tab

if (options.cookies) {

if (getCookie("page")) { showTab(getCookie("page")); }

else { setCookie("page",1,999); showTab(1); }

} else {

showTab(1);

}

function showTab(num) {

tabIndex[num].addClass("active").siblings().removeClass("active");

if(!options.animate) { tabs[num].show().siblings().hide(); }

else {

switch (options.effect) {

case "fade":

tabs[num].fadeIn(options.speed).siblings().hide();

break;

case "slide":

tabs[num].slideDown(options.speed).siblings().hide();

break;

}

}

}

obj.find("li").live("click", function(e){

e.preventDefault();

var tab\_num = $(this).attr("title");

showTab(tab\_num);

if (options.cookies) setCookie("page",tab\_num,999);

});

/\* code to handle cookies \*/

function setCookie(c\_name,value,expiredays)

{

var exdate=new Date();exdate.setDate(exdate.getDate()+expiredays);document.cookie=c\_name+"="+escape(value)+

((expiredays==null)?"":";expires="+exdate.toUTCString());

}

function getCookie(c\_name)

{

if(document.cookie.length>0)

{c\_start=document.cookie.indexOf(c\_name+"=");if(c\_start!=-1)

{c\_start=c\_start+c\_name.length+1;c\_end=document.cookie.indexOf(";",c\_start);if(c\_end==-1)c\_end=document.cookie.length;return unescape(document.cookie.substring(c\_start,c\_end));}}

return"";

}

});

};

})(jQuery);

**Db\_Shop.sql**

CREATE TABLE IF NOT EXISTS `buy\_products` (

`buy\_id` int(11) NOT NULL AUTO\_INCREMENT,

`buy\_id\_order` int(11) NOT NULL,

`buy\_id\_product` int(11) NOT NULL,

`buy\_count\_product` int(11) NOT NULL,

PRIMARY KEY (`buy\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO\_INCREMENT=3 ;

-- --------------------------------------------------------

--

-- Структура таблицы `cart`

--

CREATE TABLE IF NOT EXISTS `cart` (

`cart\_id` int(11) NOT NULL AUTO\_INCREMENT,

`cart\_id\_product` int(11) NOT NULL,

`cart\_price` int(11) NOT NULL,

`cart\_count` int(11) NOT NULL DEFAULT '1',

`cart\_datetime` datetime NOT NULL,

`cart\_ip` varchar(100) NOT NULL,

PRIMARY KEY (`cart\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO\_INCREMENT=13 ;

-- --------------------------------------------------------

--

-- Структура таблицы `category`

--

CREATE TABLE IF NOT EXISTS `category` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`type` varchar(20) NOT NULL,

`brand` varchar(50) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO\_INCREMENT=25 ;

**Index.html**

<html>

<head>

<title>CKEditor Samples</title>

<meta charset="utf-8">

<link rel="stylesheet" href="sample.css">

</head>

<body>

<h1 class="samples">

CKEditor Samples

</h1>

<div class="twoColumns">

<div class="twoColumnsLeft">

<h2 class="samples">

Basic Samples

</h2>

<dl class="samples">

<dt><a class="samples" href="replacebyclass.html">Replace textarea elements by class name</a></dt>

<dd>Automatic replacement of all textarea elements of a given class with a CKEditor instance.</dd>

<dt><a class="samples" href="replacebycode.html">Replace textarea elements by code</a></dt>

<dd>Replacement of textarea elements with CKEditor instances by using a JavaScript call.</dd>

</dl>

<h2 class="samples">

Basic Customization

</h2>

<dl class="samples">

<dt><a class="samples" href="uicolor.html">User Interface color</a></dt>

<dd>Changing CKEditor User Interface color and adding a toolbar button that lets the user set the UI color.</dd>

<dt><a class="samples" href="uilanguages.html">User Interface languages</a></dt>

<dd>Changing CKEditor User Interface language and adding a drop-down list that lets the user choose the UI language.</dd>

</dl>

**ПРИЛОЖЕНИЕ Б**

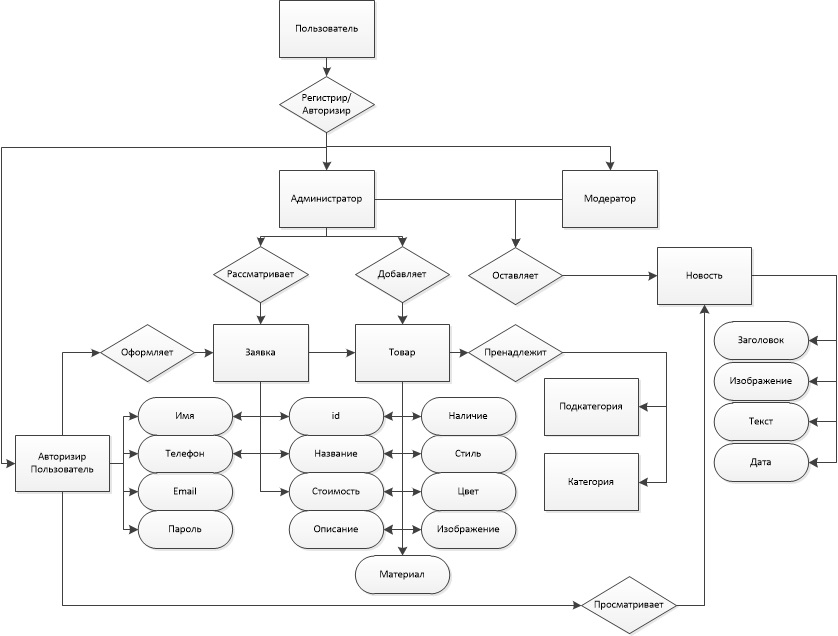


Рисунок 26 – Логическая схема