Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Институт информационных технологий БГУИР

Кафедра «Микропроцессорные системы и сети»

**К защите допустить**

Заведующий кафедрой

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

*\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_*2017 г.

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К ДИПЛОМНОЙ РАБОТЕ**

**НА ТЕМУ:**

**«Информационный сайт телевизионных приставок»**

Слушатель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Э.Л.Дударёнок

(дата, подпись)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.А.Листопадов

(дата, подпись)

Консультанты:

по ЕСПД и ЕСКД\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_П.В. Полторецкая

(дата, подпись)

по специальности \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Ф.Н. Козелько

(дата, подпись)

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата, подпись)

МИНСК 2017

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Институт информационных технологий БГУИР

Кафедра микропроцессорных систем и сетей

Специальность 1-40 01 74 «Web-дизайн и компьютерная графика»

***УТВЕРЖДАЮ***

Зав. кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Подпись, Ф.И.О.)

"\_\_\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2017 г.

**ЗАДАНИЕ**

по дипломному проектированию

Слушателю *Дударёнку Эдуарду Леонидовичу*

1.Тема проекта *Информационный сайт телевизионных приставок, утверждена приказом по Институту №123-С от 04.08.2017*

2. Сроки сдачи слушателем законченного проекта *16 октября 2017г.*

3. Исходные данные к проекту *Сайт телевизионных приставок предоставляет информацию пользователям о технических характеристиках, ценах, магазинах и т.д.*

4. Перечень подлежащих разработке вопросов, содержащихся в пояснительной записке:

*1. Анализ предметной области и составление технического задания.*

*1.1 Определение целей и задач сайта.*

*1.2 Анализ целевой аудитории.*

*1.3 Анализ сайтов конкурентов.*

*1.4 Бриф.*

*2.Проектирование сайта.*

*2.1* *Эргономика сайта.*

*2.2 Проектирование сайта.*

*2.3 Разработка структуры базы данных.*

*3. Разработка сайта.*

*4. Тестирование сайта.*

*5. Продвижение сайта.*

5. Консультанты по проекту (с указанием относящихся к ним разделов проекта)

*по специальности –Ф.Н. Козелько*

*по ЕСКД, ЕСПД –* *П.В. Полторецкая*

6. Дата выдачи задания *22.07.2017*

7. Календарный график работы над проектом на весь период проектирования

*05.08.2017г.- 30% (п.п. 1.1, 1.2, 1.3);*

*19.08.2017г.- 50% (п.п. 2.1, 2.2, 2.3);*

*05.09.2017г.–70% (п. 3);*

*21.09.2017г.–100% (п. 4, 5 и ПЗ).*

Руководитель (C*.А. Листопадов, специалист учебного отдела ИИТ)*

(подпись, Ф.И.О., должность)

Задание принял к исполнению (дата и подпись слушателя)

**СОДЕРЖАНИЕ**

|  |  |
| --- | --- |
| Введение | 4 |
| 1 Разработка технической концепции | 5 |
| 1.1 Определение целей и задач сайта | 5 |
| 1.2 Анализ целевой аудитории | 5 |
| 1.3 Анализ сайтов конкурентов | 6 |
| 1.4 Бриф | 20 |
| 2 Проектирование сайта | 23 |
| 2.1 Эргономика сайта (юзабилити) | 23 |
| 2.2 Проектирование макета сайта | 26 |
| 2.3 Проектирование структуры базы данных | 29 |
| 3 Разработка сайта | 34 |
| 3.1 Использованные технологии в проекте | 34 |
| 3.2 Принцип работы базисных классов | 35 |
| 3.3 Разработка страниц сайта | 39 |
| 4 Тестирование сайта | 49 |
| 5 Продвижение сайта | 52 |
| 5.1 SEO продвижение | 52 |
| 5.2 Разработка новых функциональных возможностей | 56 |
| Заключение | 58 |
| Список использованных источников | 59 |
| Приложение А (обязательное)Листинг кода | 60 |
|  |  |
|  |  |
|  |  |
|  |  |

**ВВЕДЕНИЕ**

В последнее время сфера телевизионного вещания подвергается настоящим революционным изменениям. С развитием сети интернет, доступность новостного, развлекательного, образовательного и иного видео контента стала лучше и проще.

Теперь для того, чтобы узнать последние новости или посмотреть очередную серию любимого сериала, совсем не обязательно изучать программу телевизионных передач и подстраивать свой режим под вещательную сетку. Достаточно просто открыть сервис потокового видео вещания вроде YouTube, Twitch.tv, Periscope или воспользоваться услугами видеохостинга, к примеру, таких, как Netflix, Ivi, Apple TV и др.

Однако, проблема в том, что большинство существующих моделей телевизоров не предназначены для просмотра видео контента из таких источников. Как правило, для просмотра требуется установка клиентской части программного обеспечения на устройство воспроизведения. Существующие сегодня телевизоры построены на весьма ограниченных, как функционально, так и технически, операционных системах, которые не предусматривают установку стороннего программного обеспечения.

Выходом из данной ситуации является приобретение внешнего устройства, которое позволяет принимать цифровой видео контент и выводить его на экран телевизора. Такие телевизионные приставки управляются операционной системой Android, что позволяет установить любое нужное приложение.

Такие устройства обходятся покупателям в сравнительно небольшие суммы. И отказ от услуг классических телевизионных провайдеров позволяет буквально за несколько месяцев компенсировать стоимость такого устройства.

Разрабатываемый мной сайт представляет собой информационный ресурс, посвященный современным телевизионным приставкам, а также проблемам выбора и покупки таких устройств.

Далее в дипломной работе я буду также использовать выражения «твбокс» и «медиабокс», которые являются заимствованными из английского языка и, которые являются синонимами для выражения «современная телевизионная приставка».

Цель дипломной работы – разработать сайт телевизионных приставок.

**1 РАЗРАБОТКА ТЕХНИЧЕСКОЙ КОНЦЕПЦИИ**

* 1. **Определение целей и задач сайта**

Целью данной работы является создание информативного, удобного и понятного для пользователей информационного сайта телевизионных приставок.

Назначение сайта — эффективная коммуникация с пользователями на разных этапах их ознакомления с технологиями современных телевизионных приставок, помощь в наиболее выгодной реализации их потребностей.

Сайт должен решать следующие задачи:

* Предоставление пользователям информации о существующих моделях популярных телевизионных приставок.
* Предоставление возможности сравнения технических характеристик различных моделей телевизионных приставок.
* Предоставление пользователям сведений о продавцах телевизионных приставок, а также актуальной ценовой информации.
  1. **Анализ целевой аудитории**

Целевая аудитория (на английском — target audience, target group) — это совокупность реальных и потенциальных потребителей товара или услуги, которые готовы изменить свои предпочтения в пользу данного товара или услуги под воздействием маркетинговых мер.

При создании нового интернет ресурса, для достижения максимального успеха в продвижении товара или услуги, необходимо точно определить целевую аудиторию (ЦА) [1].

Данный ресурс предназначен для следующих категорий пользователей:

* Пользователь, который ищет в сети интернет первичную информацию о том, что такое твбоксы, какие у них преимущества и какое назначение. Типичный представитель – молодой человек 25-35 лет, женат, образование среднее или высшее, не склонен к спонтанным покупкам, любит технологические новинки.
* Пользователь, который хочет приобрести телевизионную приставку, но не знают на какой модели остановить свой выбор. Типичный представитель – мужчина 30-40 лет, женат, образование среднее или высшее, не склонен к спонтанным покупкам, согласен заплатить дороже и потратить больше времени на изучение вопроса, но получить лучший продукт по соотношению цена качество.
* Пользователь, который столкнулся с проблемами при эксплуатации устройства и ему надо задать вопрос или обменяться мнением с другими владельцами. Типичный представитель – молодой человек 20-30 лет, женат или не женат, образование среднее или высшее, коммуникабелен, усидчив, разбирается в технических вопросах.
* Менеджер интернет магазина, который занимается реализацией телевизионных приставок. Типичный представитель – молодой человек 20-30 лет, женат или не женат, коммуникабелен, технически подкован.
  1. **Анализ сайтов конкурентов**

Анализ сильных конкурентов может быть весьма полезным при создании сайта в тактическом смысле. Проанализировав качественные сайты, можно получить новые идеи, узнать о маленьких хитростях, специфических способах продвижениях и прикольных фишках, которые повысят эффективность проекта.

Правильно сделанный анализ сайтов конкурентов позволяет избежать многих ошибок еще на этапе проектирования интернет ресурса. Для анализа были выбраны небольшие сайты, специализирующиеся на продаже или продвижении преимущественно телевизионных приставок.

Крупные коммерчески сайты каталоги, предлагающие весь спектр технических и бытовых товаров, по сути не являются сайтами конкурентами, так как пересечение целевых аудиторий минимально.

Анализ проводился по следующим критериям:

* дизайн;
* юзабилити;
* возможность сравнения цен у разных продавцов;
* возможность сравнения технических характеристик нескольких выбранных моделей;
* наличие страницы отзывов;
* возможность задать интересующий вопрос;
* актуальность и полнота представленной информации;
* технический анализ сайта.

**Интернет магазин** **INVIN** (https://smartpristavka.ru).

На рисунке 1.1 изображена главная страница интернет магазина INVIN, специализирующегося на продаже телевизионных приставок и другого мультимедийного оборудования.

### Изображение выглядит как снимок экрана Описание создано с очень высокой степенью достоверности

### Рисунок 1.1 – Главная страница сайта INVIN

Дизайн сайта построен на популярных на сегодняшний день принципах плоского дизайна. Практически полностью отсутствуют элементы, направленные на придание глубины и объемности: такие как тени, рефлексы, градиенты, текстуры, блики.

Вместе с тем, использование светлых тонов в оформлении сайта, небольшого количества цветовых акцентов и правильно подобранных шрифтов (семейство Open Sans), повышает читаемость контента. Пользователь буквально с первых мгновений посещения сайта понимает, что он находится в интернет магазине.

Анализ юзабилити сайта позволяет сделать вывод, что элементы навигации расположены в стандартных для пользователей местах и не вызывают неудобства или дискомфорта. Графический и текстовый контент подобраны исходя из особенностей целевой аудитории. На главной странице сайта отсутствует лишняя информация, напрасно перегружающая пользователя.

На рисунке 1.2 представлена страница карточки товара сайта INVIN.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.2 – Страница карточки товара сайта INVIN

На рисунке 1.2 видно, что при оформлении блока технических характеристик используются иконки. При этом иконки выполнены в разных стилях, часть иконок обладает плохой различимостью и читаемостью.

Сайт INVIN является интернет магазином и, соответственно, не размещает ссылки на других продавцов. Таким образом возможность произвести сравнение цен и предложений отсутствует.

К недостаткам сайта можно отнести тот факт, что отсутствует возможность сравнения технических характеристик разных моделей телевизионных приставок.

На сайте есть возможность оставить отзыв или задать интересующий вопрос. Для этих целей на странице любой модели есть блок «Отзывы». Авторизация пользователей не требуется, достаточно ввести только имя.

На рисунке 1.3 представлены ошибки выполнения кода на сайте INVIN.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.3 – Ошибки кода сайта INVIN

На рисунке 1.3 видно, что использование консоли разработчика в браузере chrome позволяет выявить несколько ошибок. Все четыре ошибки связаны с распространением рекламных материалов.

Информация, представленная на сайте, актуальна и соответствует действительности. Однако, модельный ряд продаваемых телевизионных приставок весьма беден. В каталоге всего 16 моделей, большинство из них выпущены в 2014-2016 годах.

Интернет магазин INVIN построен с использованием готовой CMS системы OpenCart. Выбор данной платформы безусловно оправдан, так как OpenCart изначально позиционируется, как лучшее решение для интернет магазинов с различными методами оплаты и доставки.

Благодаря применению адаптивного шаблона, сайт в целом хорошо смотрится на любой диагонали, в том числе и на мобильных устройствах. Единственным исключением является блок с промо видео на главной странице. Он плохо масштабируется на экранах шириной менее 790 пикселей.

**Интернет магазин ТВ-приставка** (https://android-tv-pristavka.ru).

Основная цель сайта ТВ-приставка – популяризация и продажа телевизионных приставок на операционной системе Android. В качестве сопутствующего товара, также предлагают приобрести пульты управления.

На рисунке 1.4 изображена главная страница магазина ТВ-приставка.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.4 – Главная страница магазина ТВ-приставка

Дизайн сайта основан на принципах Flat дизайна. Использование светлых тонов, шрифтов семейства Open Sans, а также отсутствие теней, объемов и текстур делают сайт легковесным и лаконичным.

Цветовые акценты используются умеренно, вкрапления яркого цвета приходятся только на кнопки целевых действий: «Купить» и «Добавить в корзину». Вместе с тем, отсутствие явного оформления карточек товаров на главной странице, а также близкое расположение блоков к друг другу, снижает читаемость и ухудшает восприятие контента пользователем.

В главном меню сайта используется цвет шрифта очень близкий к цвету фона, т.е. невнимательный пользователь может его вообще не увидеть.

На рисунке 1.5 представлена страница карточки товара.

Изображение выглядит как снимок экрана

Описание создано с высокой степенью достоверности

Рисунок 1.5 – Страница карточки товара сайта ТВ-приставка

Анализ юзабилити сайта позволил выявить несколько недочетов, ухудшающих опыт взаимодействия пользователя. К примеру, основной логотип выполнен в очень маленьком размере. Использование мелкого текста в логотипе делает его практически нечитаемым.

Так как сайт ТВ-приставка является интернет магазином, возможность сравнения цен у разных продавцов отсутствует. Ссылки на другие магазины, ввиду очевидной конкурентности, не размещаются. Также нет возможности выбрать несколько моделей и одновременно сравнить их технические характеристики. Такая функция была бы очень полезна для пользователей, которые плохо ориентируются в представленных на сайте предложениях.

На странице карточки товара помимо актуальной и полной технической информации, присутствует возможность скачать инструкцию пользователя, напечатанную производителем товара.

Очевидным минусом сайта является небогатый каталог товаров. Вниманию пользователей представлены только пять моделей телевизионных приставок и только два вида пультов.

Также к минусам можно отнести отсутствие сведений о наименовании юридического лица, от имени которого осуществляются продажи. В разделе «Контакты» указан только адрес и телефон.

На рисунке 1.6 можно увидеть результат проверки главной страницы сайта с помощью консоли разработчика браузера chrome.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.6 – Ошибки кода сайта ТВ-приставка

Проверка критических ошибок не выявила. Однако, браузер сформировал 81 предупреждение о том, что картинки сайта загружаются по незащищенному протоколу http.

Сайт ТВ-приставка создан с использованием CMS системы Wordpress. Применяется адаптивный шаблон и страницы ресурса хорошо выглядят на устройствах с любыми диагоналями.

**Интернет магазин Покупка5** (http://pokupka-5.ru).

Ресурс Покупка5 представляет собой интернет магазин, специализирующийся на продаже телевизионных приставок, мультимедийного оборудования и прочих современных гаджетов.

На рисунке 1.7 изображена главная страница сайта Покупка5.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.7 – Главная страница сайта Покупка5

Дизайн сайта оставляет двоякое впечатление. С одной стороны, отсутствуют явно раздражающие внимание пользователя элементы. Цветовые акценты применяются в меру и не вызывают эстетического отторжения. Используются хорошо читаемые шрифты семейства Roboto Condensed.

С другой стороны, явно отсутствует стилевое единство и общая концепция дизайна не продуманна.

На рисунке 1.8 представлена страница карточки товара сайта Покупка5.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.8 – Страница карточки товара сайта Покупка5

Из очевидных минусов дизайна и юзабилити также можно выделить следующие моменты:

* низкое разрешение логотипа сайта;
* все элементы прижаты друг к другу, им явно не хватает «воздуха»;
* навигация по сайту реализована неудобно для пользователя - в меню нет ссылки на каталог;
* анкоры ссылок очень длинные, некоторые состоят из предложений;
* карточки товаров на главной странице оформлены не информативно, для получения минимальной информации надо переходить по ссылке;
* технические характеристики телевизионных приставок отображаются не наглядно и не информативно.

На сайте Покупка5 отсутствует возможность сравнения цен с другими продавцами. Также не предусмотрена возможность одновременного сравнения технических характеристик нескольких моделей телевизионных приставок. Представление таких данных в табличном виде вообще не используется.

Несмотря на то, что на сайте есть специальная страница для отзывов покупателей, она используется только для обсуждения вопросов приобретения. Непосредственно на страницах самих товаров нет ни форм обратной связи, ни мини-чатов, ни отзывов.

Актуальность и полнота представленной информации нареканий не вызывает. Все технические характеристики представлены в полном объеме. Также представлено большое количество графического материала. Есть отдельный раздел с видео материалами.

Технический анализ сайта позволяет сделать выводы о некоторых явных недочетах:

* сайт не адаптивный и не отзывчивый («резиновый»);
* отсутствует мобильная версия ресурса;
* используются неоптимизированные изображения большого размера;
* отключено кэширование браузером.

Вместе с тем хочется отметить отсутствие ошибок выполнения кода. При запуске консоли разработчика в браузере chrome ошибки отсутствуют.

**Интернет магазин DUNE HD** (http://dune.ru).

Ресурс DUNE HD представляет собой брендированный магазин товаров торговой марки Dune. Помимо продажи телевизионных приставок, магазин предлагает также комплекс фирменного программного обеспечения, предустановленного на твбоксы.

На рисунке 1.9 представлена главная страница сайта DUNE HD.

Изображение выглядит как внутренний, монитор, компьютер, снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.9 – Главная страница сайта DUNE HD

Дизайн сайта выполнен в духе современных тенденций и использует концепцию матириал (от англ. Material Design):

* глубина соседних блоков регулируется использованием теней;
* при скроллинге меню изменяет размер и «прилипает» к верхней части экрана;
* применяются открытые списки и изящная типографика, основной шрифт PT Sans;
* пиктограммы выполнены в стиле геометрической иконографики;
* в дизайне интерфейсов используются выразительные, привлекающие внимание цвета.

Также в дизайне сайта используется много пустого пространства, что придает дополнительную легкость и прозрачность.

На рисунке 1.10 изображена страница карточки товара. Информация о товаре полная и актуальная. Имеется возможность сравнить технические характеристики нескольких моделей одновременно.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.10 – Страница карточки товара сайта DUNE HD

Ссылки на другие магазины не указаны, возможности сравнить цены нет. Также не предусмотрена возможность задать интересующий вопрос.

При запуске консоли разработчика в браузере chrome, видны некоторые ошибки выполнения кода. Java Script обработчик ожидает от пользователя прокрутки страницы. Но, даже если ее осуществить, событие не распознается браузером и скрипт не выполняется.

На рисунке 1.11 представлен результат проверки сайта с помощью консоли разработчика браузера chrome.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.11 – Ошибки кода сайта DUNE HD

Сайт DUNE HD использует свою собственную CMS систему. Одним из минусов данного выбора является отсутствие адаптивности дизайна.

**Сайт Android TVBox и MiniPC** (http://hd-mediaplayers.ru).

Последний из рассматриваемых мной сайтов конкурентов относится к совсем другому типу интернет ресурсов, чем рассмотренные ранее. Это не интернет магазин, а информационный сайт, посвященный телевизионным приставкам и мини компьютерам.

Ресурс содержит большое количество текстовых и графических материалов, посвященных медиа устройствам. Помимо подробных обзоров, публикуется информация о скидках, акциях, распродажах, а также размещаются релевантные новости.

Рассматриваемый сайт работает на CMS Wordpress. Используется классический блоговый дизайн, где центральная область занята лентой из последних добавленных записей, а в сайдбаре содержится информация о последних новостях, баннеры, реклама и т.д.

На рисунке 1.12 изображена главная страница сайта Android TVBox и MiniPC.

Изображение выглядит как снимок экрана

Описание создано с очень высокой степенью достоверности

Рисунок 1.12 – Главная страница http://hd-mediaplayers.ru

При тестировании юзабилити сайта, в глаза бросается непродуманная навигация. На сайте отсутствует каталог. Интересующую модель можно найти только либо через поиск, либо с помощью карты сайта.

Сравнить одновременно несколько различных моделей телевизионных приставок нельзя. Сравнить цены у разных продавцов также нельзя, такая информация не публикуется.

Карточка товара, как таковая отсутствует. При переходе на страницу интересующего устройства, мы фактически получаем полноценный развернутый обзор, содержащий не только описание технических характеристик, но и фотоотчеты полной разборки твбоксов, а также результаты различных тестирований.

На сайте предусмотрена возможность оставить комментарий или задать интересующий вопрос. Для этого необходимо указать свое имя и адрес электронной почты.

Технический анализ сайта грубых ошибок не выявил. Используется полностью адаптивный шаблон, что позволяет ресурсу хорошо выглядеть на любом устройстве. Используемые на сайте изображения хорошо оптимизированы. Скорость загрузки высокая.

**1.4 Бриф**

**Рабочее название:** «myTVbox».

**Область деятельности, направление:** информационный сайт, посвященный смарт медиа устройствам.

**Адреса сайтов - конкурентов:**

* https://smartpristavka.ru;
* https://android-tv-pristavka.ru;
* http://pokupka-5.ru;
* http://dune.ru;
* http://hd-mediaplayers.ru.

**География работы компании:** страны СНГ

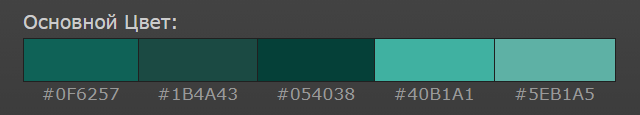
**Цели сайта:** создание информационной платформы для распространения информации о современных телевизионных приставках.

**Задачи сайта:**

* предоставление актуальной информации о твбоксах;
* предоставление возможности сравнения и анализа характеристик различных устройств;
* предоставление возможности сравнения цен различных продавцов;
* предоставление информации об актуальных скидках и распродажах;
* предоставление возможности обмена мнениями и возможности задать интересующий вопрос.

**Предполагаемый тип сайта:** информационный сайт.

**Цветовая гамма:** цветовая гамма основных цветов и дополнительных представлена на рисунке 1.13.



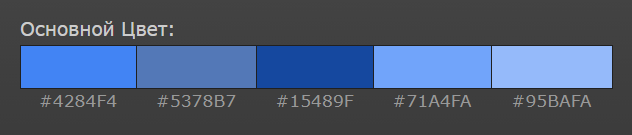


Рисунок 1.13 –Цветовая гамма

**Функциональные возможности сайта:**

* публикация технических характеристик устройств;
* публикация графических материалов (фотографии);
* публикация текстовых материалов;
* сравнение характеристик нескольких устройств;
* отправка отзывов и сообщений пользователями.

**Пользователи сайта:**

*Гость*– не авторизованный пользователь, возможности которого просмотр любого опубликованного материала.

*Резидент*– включает права гостя, а также возможность оставлять сообщения и задавать вопросы.

*Менеджер* – представитель магазина, реализующего твбоксы и заинтересованного в сотрудничестве.

*Администратор* – лицо ответственное за функционирование сайта. Обладает дополнительными правами:

* блокировка зарегистрированных пользователей;
* удаление обзоров, файлов, сообщений;
* добавление новых страниц.

UML диаграмма прецедентов сайта представлены на рисунке 1.14.



Рисунок 1.14 – Диаграмма прецедентов

**Дизайн:**

* простой, но в тоже время яркий и запоминающийся логотип;
* оригинальный стиль оформления, вызывающий ощущение технологичности;
* основной принцип дизайна - Material Design (использование анимации, теней, границ, применение карточек, создание эффекта объема);
* основной цвет белый или близкий к белому оттенок, акцентных цветов не более трех;
* короткий, но емкий слоган;
* языковая версия сайта только одна, русскоязычная;
* шрифты крупные, без засечек, минимум 6 пунктов;
* слова, характеризующие стиль сайта: технологичный, современный, продвинутый, компетентный;
* разработать лаконичный стиль информационных элементов сайта и элементов управления;
* анимационные эффекты при загрузке главной страницы;
* верстка сайта должна быть адаптивной.

**Страницы, присутствующие на сайте:**

* Главная – верхнее меню;
* Статьи – верхнее меню;
* Скидки – верхнее меню;
* Партнерам - верхнее меню;
* Сравнение медиабоксов – верхнее меню.

**Модули, присутствующие на всех страницах сайта:**

Верхняя часть сайта (шапка сайта, «header»):

* Логотип и название сайта;
* Поисковая строка;
* Верхнее меню;

Нижняя часть (Footer):

* Информация о сайте;
* Ссылки на самые популярные статьи;
* Ссылки на социальные сети;
* Ссылки на магазины партнёров.

**2 ПРОЕКТИРОВАНИЕ САЙТА**

**2.1 Эргономика сайта (юзабилити)**

При разработке сайта нельзя было не уделить внимание такому важному вопросу, как эргономика сайта. Одним из важнейших критериев оценки качества готового продукта, является дружественность интерфейса по отношению к пользователю.

Многие разработчики веб-сайтов придерживаются такого мнения, что функциональное удобство сайта имеет приоритетный характер, а эстетическая привлекательность является всего лишь желательным свойством. Однако, практика показывает, что между красотой и удобством интерфейса существует самая непосредственная связь. Ничем не примечательный с эстетической точки зрения сайт может вообще не найти отклика у современного пользователя.

Создание сайта с учетом эргономики может быть определено как способность эффективно реагировать на потребности пользователей и обеспечивать им комфорт при просмотре страницы.

В общем смысле слова термин эргономика - это использование научных знаний о человеке (психологии, физиологии, медицины) с целью улучшения условий работы на рабочем месте.

Эргономика в целом характеризуется двумя принципами:

* комфорт во время использования, который состоит из уменьшения физической и психологической усталости;
* безопасность, которая предполагает выбор подходящих решений для защиты пользователя [2].

Таким образом, проектирование пользовательского интерфейса сайта производилось с учетом вышеизложенных требований. В процессе создания сайта были определены следующие основные критерии эргономики:

* Лаконичность. На страницах ресурса не должно быть лишней информации, перегружающей пользователей. Умеренное использование анимации, осмысленная и оправданная вставка видео контента, отсутствие всплывающих окон.
* Структурированность. Информация на главной странице должна быть максимально структурирована.
* Ясность. Печатный текст намного труднее читать с монитора, чем с бумаги. Таким образом, текст должен быть достаточно разграничен. Должны использоваться шрифты без засечек.
* Персонализация информации. Публикации должны иметь такие данные, как имя автора, дату последнего обновления и дату публикации.
* Однородность структуры. Элементы, используемые для просмотра должны быть расположены на том же месте на каждой странице и должны иметь те же формы презентации с одной страницы на другую.
* Скорость загрузки и оптимизация изображений. Время загрузки страницы должно быть как можно короче. Изображения должны быть оптимизированы по размеру и эффективности сжатия.
* Адаптивность. Адаптивность характеризуется способностью веб-сайта автоматически подстраиваться под диагональ устройства пользователя.
* Цветовой контраст. Между цветом фона и текстом должен быть достаточный контраст, чтобы слабовидящие люди могли прочитать текст.
* Взаимодействие. Должна присутствовать возможность получения обратной связи от пользователей. Необходимо наличие на сайте полной контактной информации: адреса и телефонных номеров.
* Полезность. Сайт должен давать однозначные ответы на искомые вопросы пользователей.
* Навигация. Наличие на всех страницах сайта основной навигации (верхнего меню). Доступность навигации в любой момент времени без прокручивания страницы.
* Поиск. Наличие на сайте возможности поиска информации и ее доступность пользователю.

Учитывая описанные выше критерии эргономики, мной была составлена следующая структура страниц сайта.

На рисунке 2.1 представлена структура проекта.

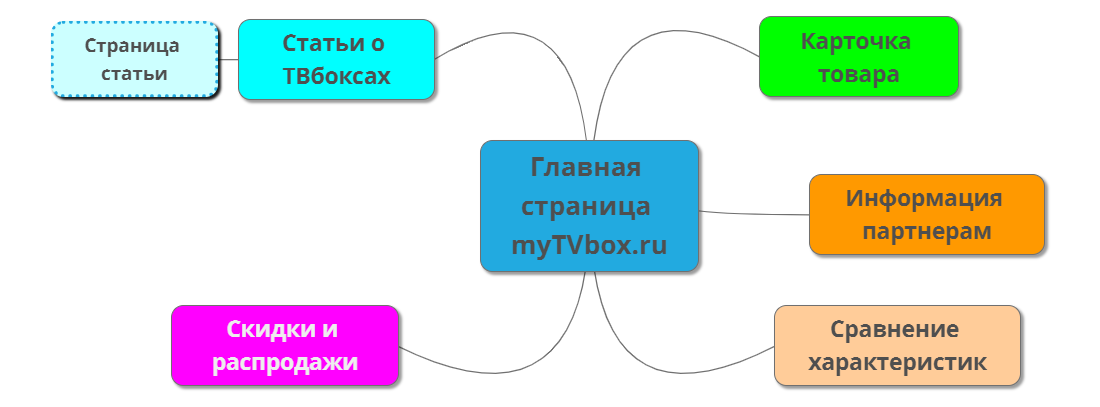


Рисунок 2.1 – Структура проекта

Навигация по страницам сайта в большинстве случаев осуществляется с помощью главного меню. Доступность навигации пользователю в любое время без прокручивания страницы обеспечено использованием «прилипающего» меню. В качестве альтернативного способа навигации выступает строка поиска.

В связи с тем, что компоновка главной страницы не предполагает размещения большого количества текстовой информации, а также в связи с необходимостью применения карточного дизайна, мной был выбран способ создания визуальной иерархии при помощи Z-паттерна.

Как понятно из названия, Z-паттерн следует форме буквы Z. Дизайн, основанный на Z-паттерне следует пути, по которому движется взгляд пользователя во время просмотра страницы – слева направо, сверху вниз.

На рисунке 2.2 представлен пример движения глаза пользователя.



Рисунок 2.2 – Пример иерархии при z-паттерне

Важные элементы располагаются на линии просмотра и таким образом формируется поток. Поток переводит взгляд пользователя от одной части контента к другой, следуя заданному направлению. Поток создается посредством комбинирования визуального веса и визуального направления.

Такой способ визуальной иерархии является очень популярным и применяется на большом количестве современных веб-сайтов. На мой взгляд, такое решение и в моем случае является наиболее уместным и оправданным.

Взгляд пользователя будет перемещаться сначала по линии верхнего меню, затем в обратном направлении по первому ряду карточек товара. После этого последует еще одна итерация и пользователь пройдет взглядом по второй линии карточек и по меню в обратном движении.

* 1. **Проектирование макета сайта**

С учетом описанных выше критериев эргономики и выбранного паттерна размещения контента, были спроектированы 6 типов страниц.

Макет главной страницы представлен на рисунке 2.3.

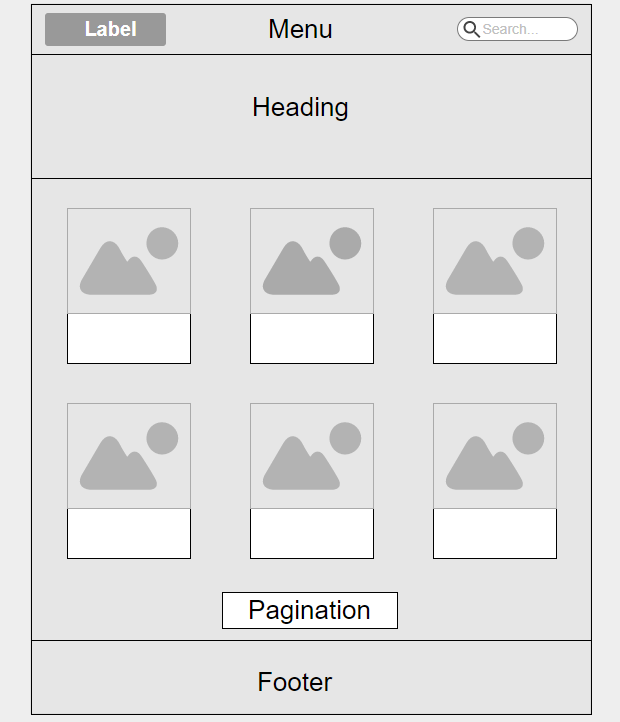


Рисунок 2.3 – Макет главной страницы сайта

Основная задача главной страницы сайта – это дать пользователю первоначальное представление о том, что такое современная телевизионная приставка на операционной системе Андроид, а также познакомить с некоторыми конкретными примерами моделей.

Главная страница представляет собой типичный мини-каталог. В верхней части расположено главное меню, затем идет баннер со слоганом, затем блок товарных карточек. Для целей оптимизации скорости загрузки страницы, количество отображаемых карточек товаров составляет шесть штук. Далее идет блок пагинации (от англ. pagination – разбивка на страницы) и в самом внизу страницы расположен футер.

Макет представления контента изображён на рисунке 2.4. Данная страница представляет собой карточку конкретной модели телевизионной приставки.

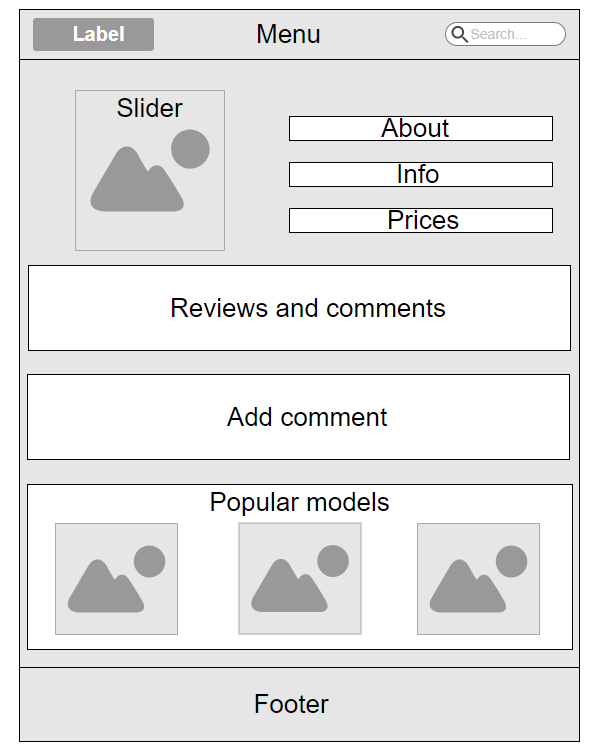


Рисунок 2.4 – Макет страницы карточки твбокса

Верхнюю часть страницы занимает меню. Ниже расположен описательный блок. В левой его части находится слайдер с картинками, а в правой аккордеон с описанием, техническими характеристиками и ценами.

Далее идет блок с отзывами и комментариями. За ним следует форма для комментария. Последний информационный блок перед футером – карточки самых популярных телевизионных приставок.

На рисунке 2.5 изображён макет сводной страницы статей, а также макет отдельной статьи.

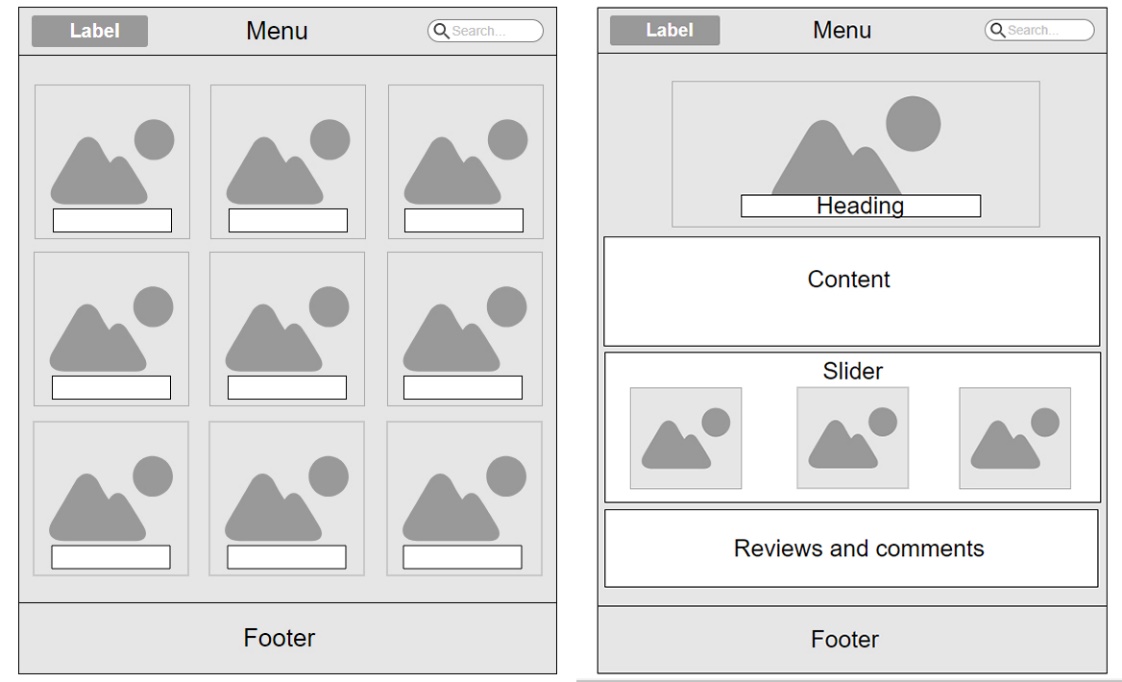


Рисунок 2.5 – Макеты сводной страницы и макет отдельной статьи

Сводная страница всех статей также построена в карточном стиле. Макет этой страницы предполагает размещение 9 статей, при большем количестве контента будет добавлен блок пагинации.

Макет страницы конкретной статьи построен следующим образом. Под блоком главного меню находится заголовок, размещенный на фоне сопровождающего изображения. Далее идет контентный блок, за ним слайдер со ссылками на другие статьи, а в самом конце блок комментариев и футер.

Макет страницы сравнения и страницы с контактами изображён на рисунке 2.6.

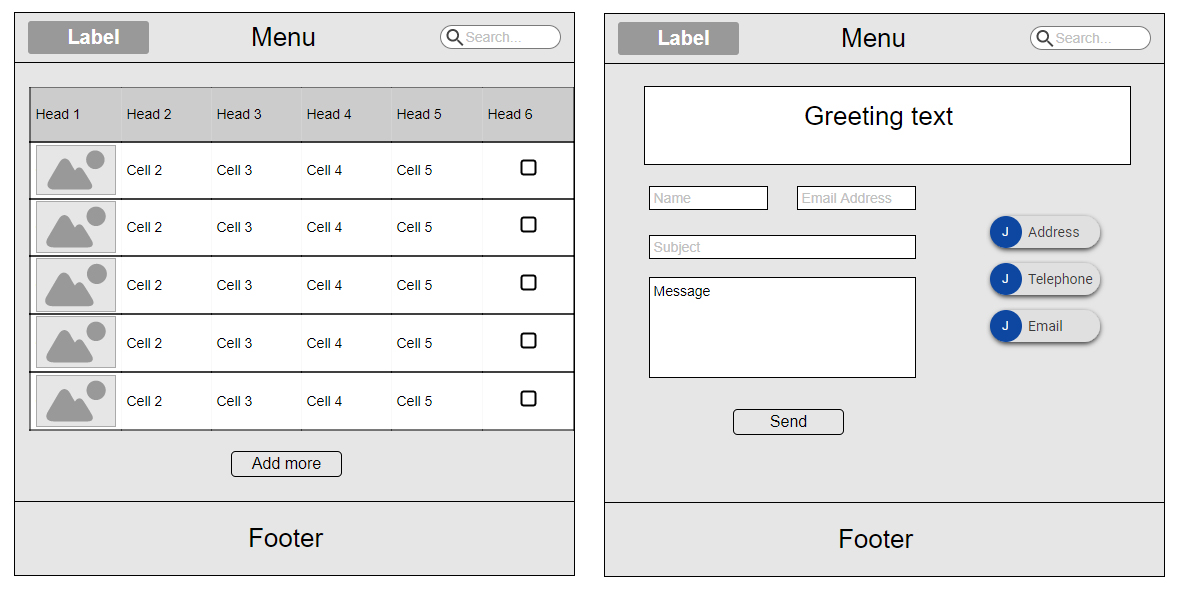


Рисунок 2.6 – Макет страницы сравнения и страницы контактов

Страница сравнения приставок представляет собой сводную таблицу технических характеристик разных моделей.

Макет страницы контактов предполагает размещение текстового блока, формы для отправления письма и блока контактных данных представителей ресурса.

* 1. **Проектирование структуры базы данных**

Базы данных используются повсюду, включая большую часть проектов в мире веб-разработки. Всё, начиная от простейших блогов и каталогов, до серьезных социальных веб-проектов. Независимо от сложности сайта и соответствующей базы данных, каждый из них требует тщательного проектирования, чтобы работать эффективно, а также надежно.

Проектирование баз данных - процесс создания схемы базы данных и определения необходимых ограничений целостности.

Основные задачи проектирования баз данных:

* обеспечение хранения в БД всей необходимой информации;
* обеспечение возможности получения данных по всем необходимым запросам;
* сокращение избыточности и дублирования данных;
* обеспечение целостности базы данных.

Основная цель проектирования базы данных - это сокращение избыточности данных, уменьшение затрат на многократные операции обновления избыточных копий и устранение возможности возникновения противоречий из-за хранения в разных местах сведений об одном и том же объекте [3].

Процесс проектирования БД представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае можно выделить следующие этапы проектирования:

* Системный анализ и словесное описание информационных объектов предметной области.
* Проектирование инфологической модели предметной области — частично формализованное описание объектов предметной области в терминах некоторой семантической модели.
* Даталогическое или логическое проектирование БД, то есть описание БД в терминах принятой диалогической модели данных.
* Физическое проектирование БД, то есть выбор эффективного размещения БД на внешних носителях для обеспечения наиболее эффективной работы приложения.

На первом этапе я провел подробное словесное описание объектов предметной области и реальных связей, которые присутствуют между описываемыми объектами. При определении состава и структуры предметной области использовался функциональный подход*.*

Такой подход применяется, когда необходимо реализовать принцип движения «от задач» и применяется тогда, когда заранее известны функции и задачи, для обслуживания информационных потребностей которых создается рассматриваемая БД.

Следующим этапом процесса проектирования стало построение инфологической модели данных, т. е. идентификации сущностей. Затем, на стадии проектирования даталогической модели базы данных, инфологическая модель была отображена в компьютерно-ориентированную даталогическую модель, «понятную» СУБД.

При проектировании базы данных сайта, в целях повышения эффективности работы и удобства обращения с БД, мной были учтены рекомендации реляционной модели данных по нормализации структуры базы.

Нормальная форма (НФ)— требование, предъявляемое к структуре таблиц в теории реляционных баз данных для устранения из базы избыточных функциональных зависимостей между атрибутами (полями таблиц).

Процесс проектирования БД с использование метода нормализации отношений является итерационным и заключается в последовательном переводе отношения из 1НФ в НФ более высокого порядка по определенным правилам. Каждая следующая НФ ограничивается определенным типом функциональных зависимостей и устранением соответствующих аномалий при выполнении операций над отношениями БД, а также сохранении свойств, предшествующих НФ [4].

Таким образом, учитывая вышеизложенное, спроектированная мной структура базы данных, была максимально приведена к пяти нормальным формам.

***Отношение находится в 1НФ.*** Все атрибуты являются простыми, все используемые домены содержат только скалярные значения. Повторения строк в таблице отсутствуют.

***Отношение находится во 2НФ.*** Структура БД находится в 1НФ и каждый неключевой атрибут функционально полно зависит от первичного ключа. Это означает, что в каждой таблице хранятся только те данные, которые напрямую связаны с ней и не имеют отношения к другой сущности

***Отношение находится в 3НФ.*** Структура БД находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Т.е. все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы, вынесены в отдельные таблицы.

***Отношение находится в 4НФ.*** Структура БД находится в 3НФ и все нетривиальные многозначные зависимости фактически являются функциональными зависимостями от ее потенциальных ключей.

***Отношения находятся в 5НФ.*** Структура БД находится в 4НФ и сложные зависимые соединения между атрибутами отсутствуют.

Учитывая всё выше сказанное, структура спроектированной базы данных приведена на рисунке 2.7.

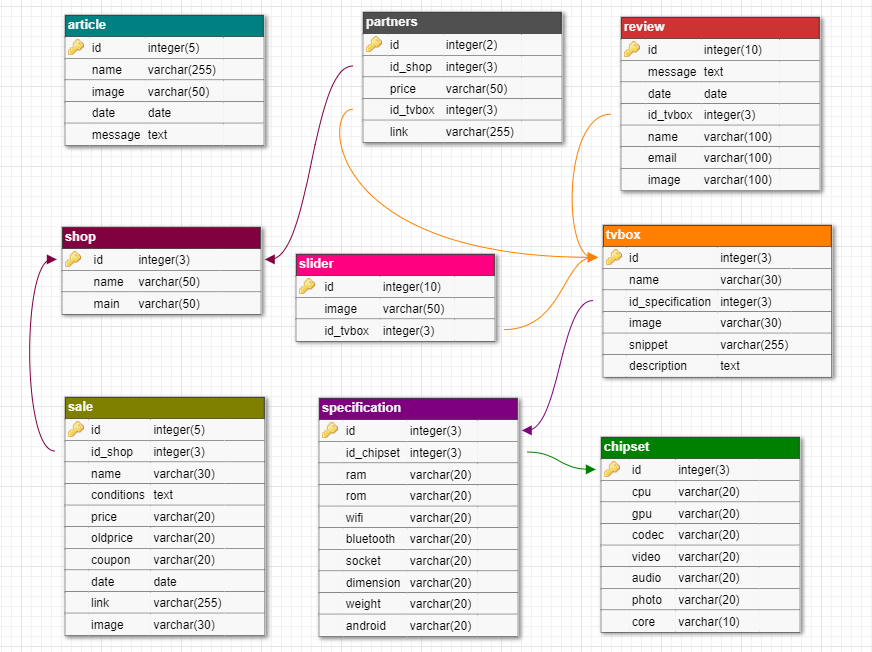


Рисунок 2.7 – Структура базы данных проекта

В каждой таблице поле id помечено иконкой замочного ключа и это означает первичный ключ таблицы. Для внешних ключей используются названия вида id\_\*, где \* означает название связанной таблицы.

***tvbox*** – таблица данных телевизионных приставок.

* name – название модели;
* id\_specification – номер в таблице с техническими характеристиками (specification);
* image – картинка, используемая для оформления карточки товара на главной странице сайта;
* snippet – текст карточки товара на главной странице;
* description – тестовое описание твбокса, отображаемое на странице конкретной модели.

***partners*** – таблица с ценами и ссылками на магазины партнеры.

* id\_shop – номер магазина в таблице с магазинами партнерами (shop);
* price – цена приставки в магазине партнере;
* id\_tvbox – номер приставки в таблице моделей (tvbox);
* link – ссылка на страницу с товаром в магазине партере.

***shop*** – таблица магазинов партнеров.

* name – название магазина;
* main – ссылка на главную страницу магазина.

***slider*** – таблица в которой хранятся ссылки на изображения, используемые в слайдере на странице конкретной модели.

* image – имя файла изображения;
* id\_tvbox – номер модели приставки из таблицы твбоксов (tvbox).

***specification*** – в данной таблице хранятся часть технических характеристик телевизионных приставок. Остальные характеристики, зависящие от модели процессора устройства, вынесены в таблицу chipset.

* id\_chipset – номер в таблице процессоров (chipset);
* ram – объем оперативной памяти устройства;
* rom – объем накопителя;
* wifi – стандарт wifi адаптера устройства;
* bluetooth – стандарт bluetooth адаптера устройства;
* socket – набор портов и разъемов;
* dimension – размеры устройства;
* weight – вес устройства;
* android – версия операционной системы android.

***chipset*** – таблица технических характеристик, зависящих от модели используемого процессора.

* cpu – название и номер модели процессора;
* gpu – название и номер графического ускорителя;
* codec – поддерживаемые форматы видео сжатия;
* video – поддерживаемые видео форматы;
* audio – поддерживаемые аудио форматы;
* photo – поддерживаемые форматы изображений;
* core – количество ядер в процессоре.

***review*** – комментарии, вопросы, отзывы оставляемые пользователями на страницах приставок.

* massage – текст оставленного отзыва;
* date – дата отзыва;
* email – почтовый адрес;
* image – аватарка пользователя;
* name – имя или псевдоним пользователя;
* id\_tvbox – номер в таблице моделей приставок (tvbox).

***article*** – таблица, в которой хранятся данные информационных статей.

* name – название статьи;
* image – основная картинка используемая для оформления статьи;
* date – дата публикации статьи;
* message – контентное наполнение статьи.

***sale*** – содержит информацию о распродажах и акциях, проводимых интернет магазинами.

* id\_shop – номер магазина в таблице shop;
* name – название модели телевизионной приставки;
* conditions – условия распродажи;
* price – акционная цена твбокса;
* oldprice – старая цена пристаки;
* coupon – код скидочного купона;
* date – дата опубликования;
* link – ссылка на страницу распродажи;
* image – изображение предмета распродажи.

***shop*** – содержит информацию об интернет магазинах

* name – название магазина;
* main – ссылка на главную страницу ресурса.

1. **РАЗРАБОТКА САЙТА**
   1. **Использованные технологии в проекте**

Выбор технологии программирования и разработки — отправная точка в реализации любого web-проекта. От этого зависит функциональность, удобство будущего ресурса, то, насколько эффективным будет продвижение сайта, а также целый ряд других параметров, от быстродействия до корректности отображения на разных устройствах.

После проведенной работы по созданию макетов ресурса, а также работ по проектированию базы данных, пришло время выбрать правильные «инструменты», при помощи которых будет проводиться разработка проекта.

В качестве языков программирования, библиотек и фреймворков были выбраны:

* HTML5 в качестве стандартизированного языка разметки веб документов.
* CSS3 в качестве формального языка описания внешнего вида документа.
* JavaScript – полноценный динамический язык программирования, который применяется к HTML документу, и может обеспечить динамическую интерактивность на веб-сайтах. В данном проекте используется для анимации и манипуляции элементами DOM.
* jQuery (v3.2.1) - библиотека JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML. Библиотека jQueryпомогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Также библиотека jQuery предоставляет удобный API для работы с AJAX.
* PHP (v5.6) – язык программирования, специально разработанный для написания web-приложений (сценариев), исполняющихся на Web-сервере. Обеспечивает сборку шаблона страницы, взаимодействие с базой данных, проверку данных пришедших от пользователя.
* SQL – язык структурированных запросов, непроцедурный язык программирования, применяемый для создания, модификации и управления данными в произвольной [реляционной базе данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D1%8B%D0%B5_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляемой соответствующей системой управления базами данных ([СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%A3%D0%91%D0%94)).
* MySQL (v5.6) - свободная реляционная система управления базами данных.
* Apache HTTP-сервер (v2.4) - полнофункциональный, расширяемый веб-сервер, распространяющийся с открытым исходным кодом.
* Bootstrap (v4.0) - фреймворк для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.
* YII2 (v2.0.12) - современный PHP фреймворк для быстрой разработки веб приложений. Он позволяет максимально применить концепцию повторного использования кода и может существенно ускорить процесс веб-разработки.
* MDB (v4.4.1) – библиотека HTML- и CSS-компонентов для фреймворка Bootstrap (v4.0), написанная в духе концепции Material Design.
* CKeditor (v4) - свободный WYSIWYG-редактор кода сайта.
* ElFinder (v2.1.28) — свободный файловый менеджер для сайта.

**3.2 Принцип работы базисных классов**

При проектировании структуры веб-приложения, мной был использован архитектурный паттерн MVC(Model, View, Controller) [5].

MVC — это конструкционный шаблон, который описывает способ построения структуры приложения, сферы ответственности и взаимодействие каждой из частей в данной структуре.

Этот фундаментальный паттерн нашел применение во многих технологиях, дал развитие новым технологиям и каждый день облегчает жизнь разработчикам.

Впервые паттерн MVC появился в языке SmallTalk. Разработчики должны были придумать архитектурное решение, которое позволяло бы отделить графический интерфейс от бизнес-логики, а бизнес-логику от данных. Таким образом, в классическом варианте, MVC состоит из трех частей, которые и дали ему название.

**Модель (Model)**. Под Моделью, обычно понимается часть, содержащая в себе функциональную логику приложения. Модель должна быть полностью независима от остальных частей продукта. Модельный слой ничего не должен знать об элементах дизайна, и каким образом он будет отображаться. Достигается результат, позволяющий менять представление данных, то как они отображаются, не трогая саму Модель.

Модель обладает следующими признаками:

* модель обладает знаниями о себе самой и не знает о контроллерах и представлениях;
* для некоторых проектов модель — это просто слой данных (DAO, база данных, XML-файл);
* для других проектов модель — это менеджер базы данных, набор объектов или просто логика приложения.

**Представление (View).** В обязанности Представления входит отображение данных полученных от Модели. Однако, представление не может напрямую влиять на модель. Можно говорить, что представление обладает доступом «только на чтение» к данным.

Представление обладает следующими признаками:

* в представлении реализуется отображение данных, которые получаются от модели любым способом;
* в некоторых случаях, представление может иметь код, который реализует некоторую бизнес-логику.

На рисунке 3.1 представлена модель архитектуры паттерна MVC.

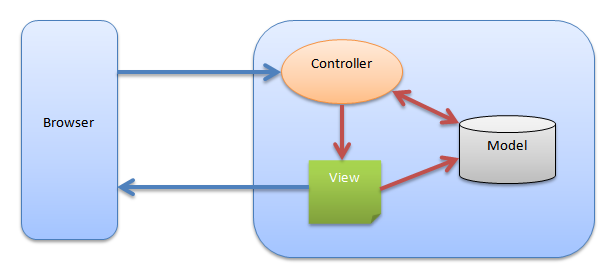


Рисунок 3.1 – Модель MVC

**Контроллер (Controller).** Контроллер обеспечивает «связи» между пользователем и системой. Контролирует и направляет данные от пользователя к системе и наоборот. Использует модель и представление для реализации необходимого действия.

Контроллер получает запросы пользователей с использованием определенных методов передачи данных (как правило GET или POST), когда пользователь нажимает на элементы интерфейса для выполнения различных действий. Его основная функция — вызывать и координировать действие необходимых ресурсов и объектов, нужных для выполнения действий, задаваемых пользователем.

Специфические признаки контроллера:

* контроллер определяет, какие представление должно быть отображено в данный момент;
* события представления могут повлиять только на контроллер;
* контроллер может повлиять на модель и определить другое представление;
* возможно несколько представлений только для одного контроллера.

Контроллер перехватывает событие извне и в соответствии с заложенной в него логикой, реагирует на это событие изменяя Mодель, посредством вызова соответствующего метода. После изменения Модель использует событие о том, что она изменилась, и все подписанные на это события Представления, получив его, обращаются к Модели за обновленными данными, после чего их и отображают.

При разработке приложения мной был использован PHP фреймворк Yii. В этом фреймворке как раз реализован описанный выше паттерн MVC. В директории models находятся Модели, в views расположены Представления, а в каталоге controllers все Контроллеры приложения.

Рисунок 3.2 демонстрирует внутреннее устройство приложения.

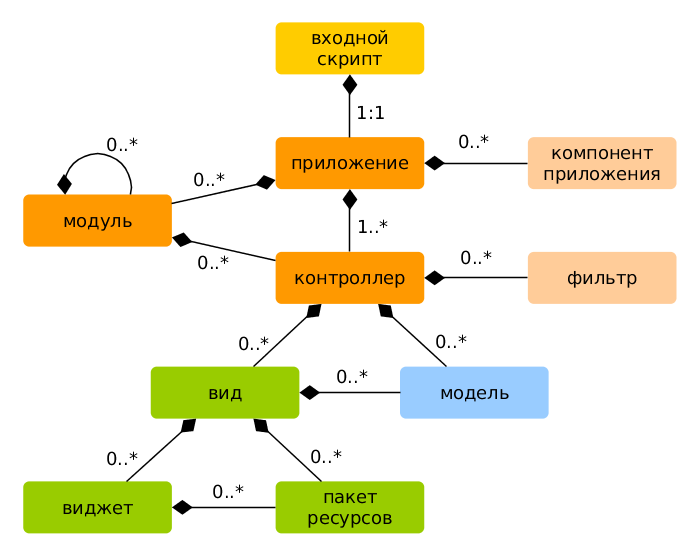


Рисунок 3.2 – Внутренне устройство приложения

Точка входа в приложение является файл web/index.php. Это единственный доступный для пользователя PHP-скрипт.

Этот скрипт принимает входящий запрос и создает экземпляр приложения. Приложение обрабатывает входящие запросы при помощи компонентов и отправляет запрос контроллеру. Виджеты используются в Представлениях для построения динамических интерфейсов сайта.

На рисунке 3.3 схематично показано, как приложение обрабатывает входящий запрос.

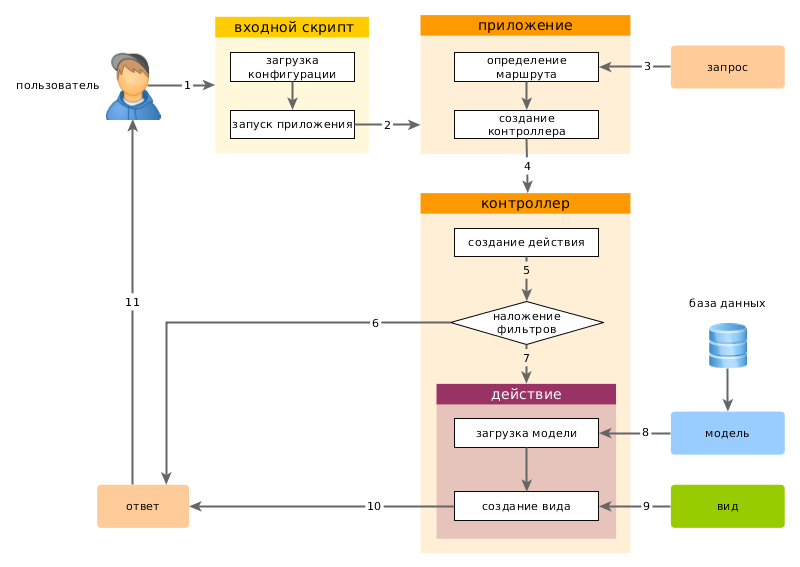


Рисунок 3.3 – Обработка запроса в приложении

Жизненный цикл пользовательского запроса выглядит следующим образом [6]:

* Пользователь обращается к точке входа web/index.php.
* Скрипт загружает конфигурацию configuration и создает экземпляр приложения для дальнейшей обработки запроса.
* Приложение определяет маршрут запроса при помощи компонента приложения «запрос».
* Приложение создает экземпляр контроллера для выполнения запроса.
* Контроллер, в свою очередь, создает действие и накладывает на него фильтры.
* Если хотя бы один фильтр дает сбой, выполнение приложения останавливается.
* Если все фильтры пройдены - приложение выполняется.
* Действие загружает модель данных. Вероятнее всего из базы данных.
* Действие генерирует вид, отображая в нем данные (в т.ч. и полученные из модели).
* Сгенерированный вид приложения передается как компонент «ответ».
* Компонент «ответ» отправляет готовый результат работы приложения браузеру пользователя.

**3.3 Разработка страниц сайта**

**3.3.1** Разработка главной страницы сайта.

Главная страница сайта, изображена на рисунке 3.4.

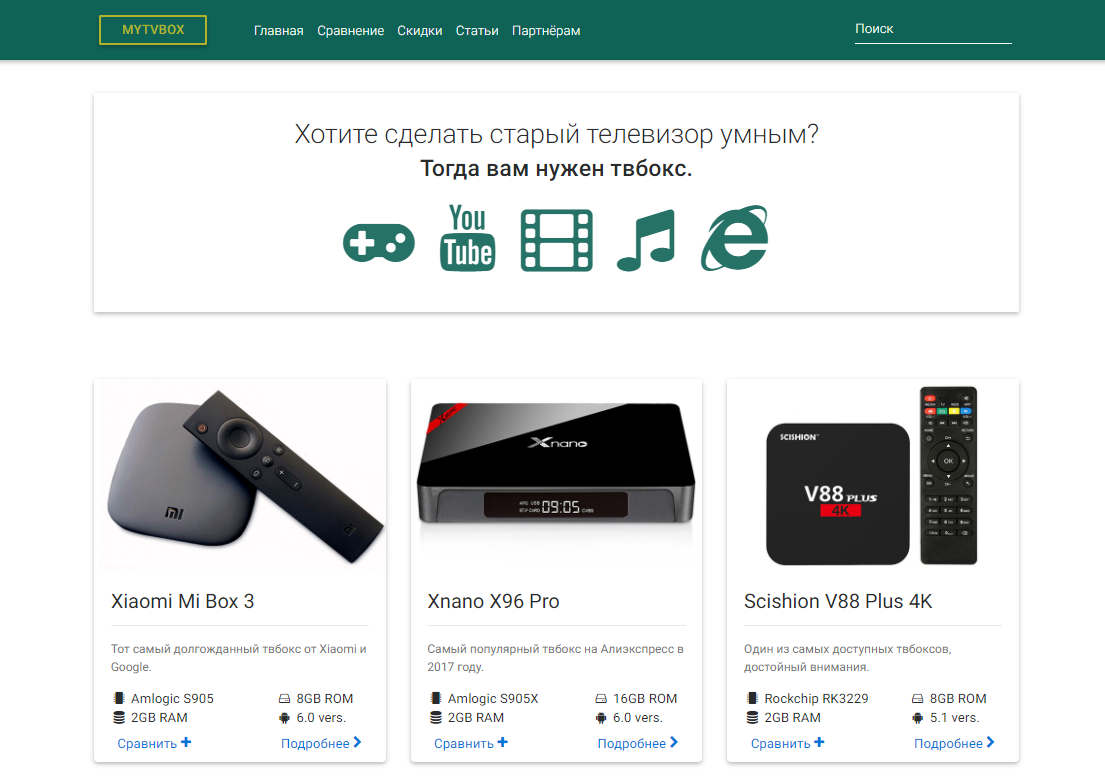


Рисунок 3.4 – Главная страница

Контроллером главной страницы является класс SiteController. Он наследуется от класса AppController, который в свою очередь наследуется от встроенного в фреймворк Yii базового класса Controller.

Страница состоит из следующих блоков:

* верхнее меню с поисковой строкой;
* баннер;
* контентная часть, содержащая шесть миниатюр карточек товаров;
* футер.

Контроллер AppController также является родительским классом и для всех остальных используемых мной в дальнейшем классов. Он имеет два метода:

* setMeta – устанавливает для формируемых страниц мета-тэги title, description, keywords;
* actions – вызывает экшен error при срабатывании ошибки.

В целях уменьшения количества кода приложения, верхнее меню и футер вынесены в отдельный лэйаут (англ. Layout – выкладка, макет, размещение) main.php и на всех страницах сайта формируются из этого файла.

Вся остальная контентная часть главной страницы содержится в представлении index.php и выводится с помощью метода actionIndex контроллера SiteController.

Блок с баннером представляет собой статичную верстку. Блок с миниатюрами карточек товаров формируется динамически из базы данных с помощью моделей Tvbox.php, Specification.php и Chipset.php.

Данные модели устанавливают необходимые связи между таблицами базы данных tvbox, specification, chipset и позволяют контроллеру совершать к ним запросы.

Контроллер получает данные из таблиц, вызывает представление index.php и передает туда в качестве ассоциативного массива полученные данные.

Представление содержит html код миниатюры карточки товара. С помощью цикла foreach из переданного ассоциативного массива генерируется необходимое количество миниатюр.

На рисунке 3.5 изображена диаграмма, отражающая действия пользователя на главной странице, а также в карточке товара.

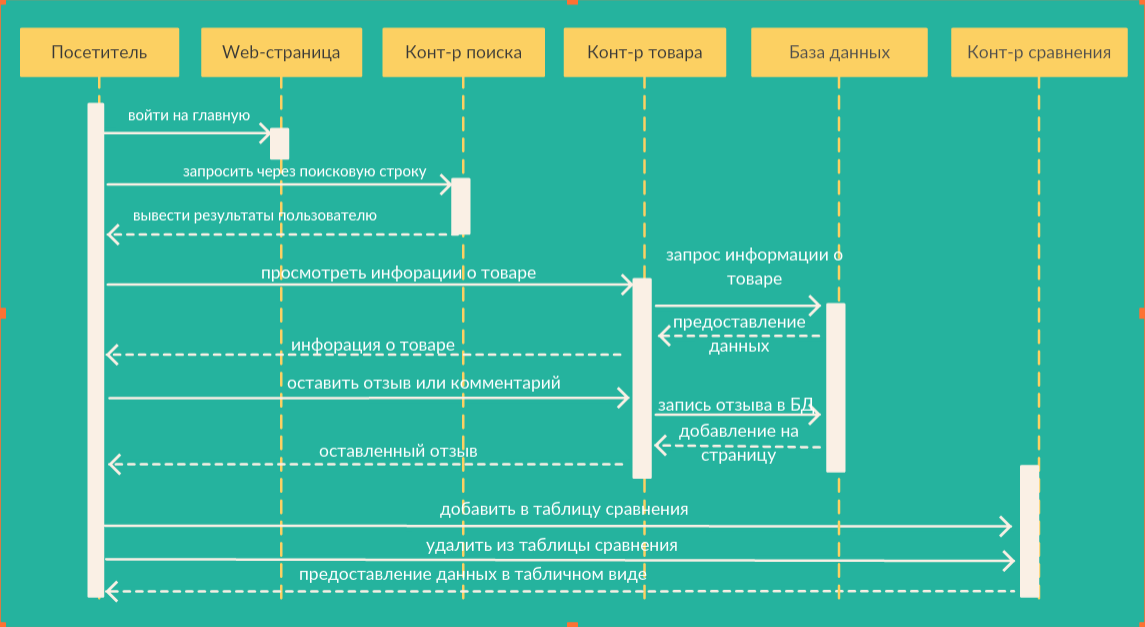


Рисунок 3.5 – Диаграмма последовательности

Исходный код упомянутых здесь и далее контроллеров, моделей и представлений приведён в приложении А.

* + 1. Разработка страницы сайта «Карточка товара».

На рисунке 3.6 изображена страница карточки товара.

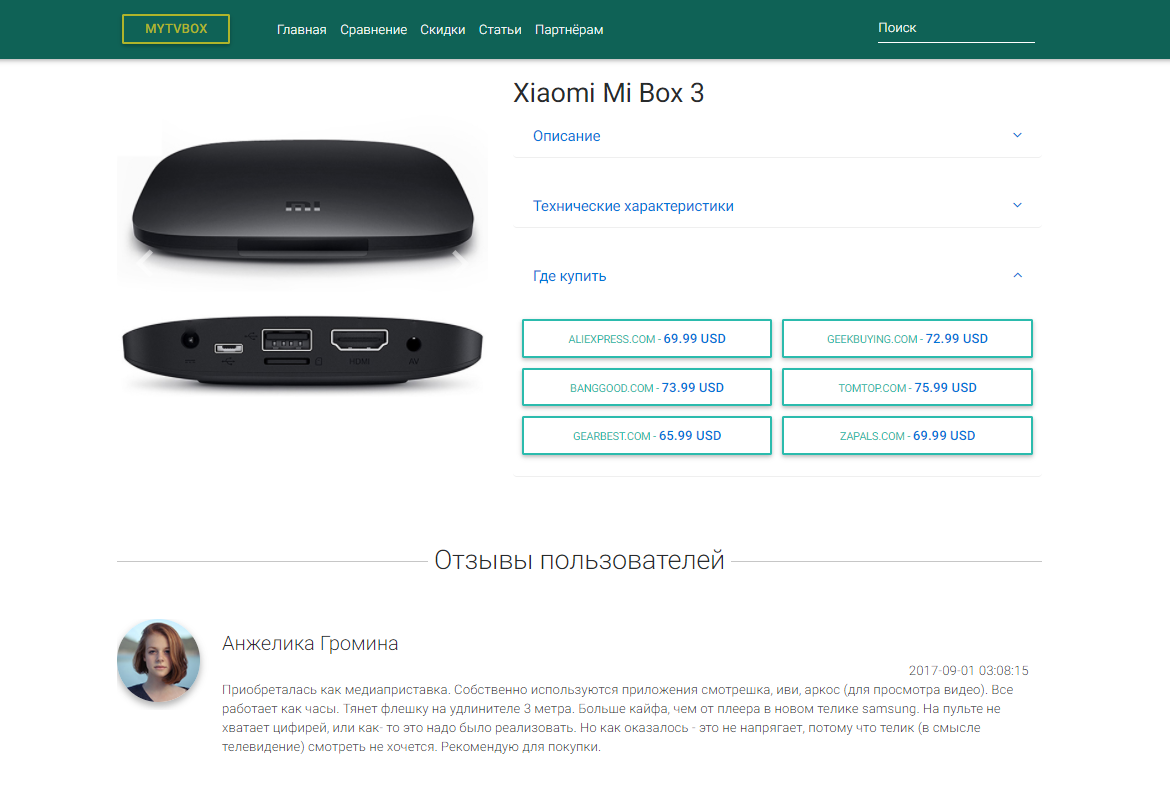


Рисунок 3.6 – Страница «Карточка товара»

На этой странице пользователь может ознакомиться с основными характеристиками товара, почитать отзывы, оставить комментарий и узнать цены на этот товар в магазинах партнёрах.

Логически страницу можно поделить на несколько блоков. В верхней левой части экрана находится слайдер с картинками, в верхней правой - аккордеон с описанием, техническими характеристиками и ценами.

Далее идёт блок с отзывами пользователей. Следом расположена форма для того, чтобы оставить новый отзыв. Внизу страницы расположен информационный блок о самых популярных моделях телевизионных приставок.

Контроллером данной страницы является также SiteController. В связи с тем, что и главная страница, и страница карточки товара получают данные о технических характеристиках из одних и тех же таблиц, было принято решение не создавать еще один контроллер, а воспользоваться уже существующим.

При переходе с главной страницы на станицу интересующей модели, пользователь должен кликнуть по ссылке «Подробнее». В момент клика в контроллер с помощью метода Get передается id параметр соответствующего твбокса.

По полученному id телевизионной приставки, контроллер c помощью метода actionBox делает запрос в базу данных и получает следующую информацию:

* из модели Tvbox – наименование товара и его описание;
* из моделей Specification, Chipset – информацию о технических характеристиках;
* из моделей Partners, Shop – информацию о ценах и ссылки на товары в магазинах партнерах;
* из модели Slider – имена картинок для слайдера;
* из модели Review – тексты отзывов посетителей, а также их имена, аватарки и даты.

Полученные из базы данные, c помощью метода render базового класс Controller, в виде ассоциативного массива передаются в представление box.php.

В представлении данные для слайдера и аккордеона подставляются на свои места, а блок отзывов генерируется динамически с помощью цикла foreach.

Блок с формой для написания отзыва позволяет пользователю без предварительной регистрации оставить комментарий. Для этого ему необходимо указать свою почту, имя, написать само сообщение и прикрепить аватарку. Все поля являются обязательными, кроме поля загрузки картинки.

Форма отправляет данные на сервер с помощью модели ReviewForm. Данная модель наследуется от базового класса ActiveRecord. С помощью метода upload модель позволяет сохранять в указанную директорию загруженные пользователем изображения.

Контроллер с помощью метода save производит не только сохранение отзывов в базу данных, но и их валидацию. Что в свою очередь предупреждает внедрения кода в БД и другие возможные атаки пользователей.

Блок с информацией о самых популярных телевизионных приставках представляет собой статичную верстку.

* + 1. Разработка страницы сравнения характеристик.

Страница «Сравнение» предназначена для вывода технических характеристик нескольких телевизионных приставок одновременно.

На рисунке 3.7 изображен пример вывода характеристик четырех твбоксов.

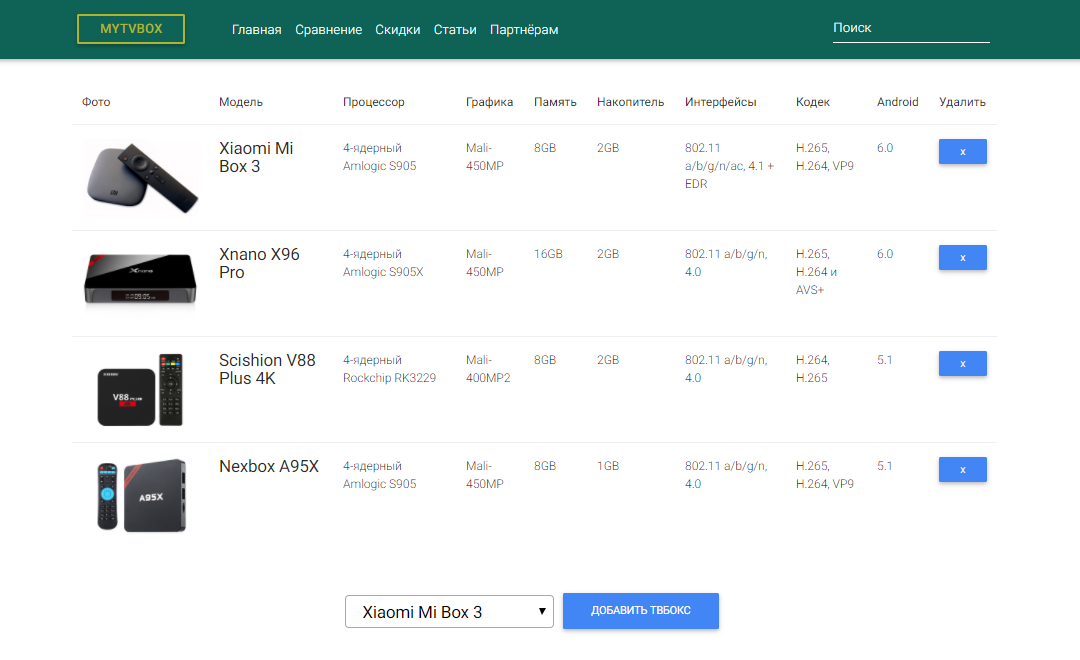


Рисунок 3.7 – Страница сравнения характеристик

Страница состоит из двух блоков, непосредственно самой таблицы и формы для добавления интересующей модели в таблицу. Второй способ добавления – ссылка «сравнить» в миниатюре карточки товара на главной странице сайта.

Добавление в таблицу модели с главной страницы осуществляется с помощью технологии AJAX. Посредством метода GET, контроллеру SiteController отправляется id параметр конкретного твбокса. Контроллер в свою очередь используя метод actionAdd, обращается к моделям Tvbox, Specification и Chipset и получает технические характеристики из базы данных.

После этого происходит обращение к модели Specs и все необходимые для отображения в таблице данные записываются в сессию с помощью метода addToSpecs.

В представлении compare.php данные извлекаются из сессии и с помощью цикла foreach формируются строки таблицы с характеристиками выбранных ранее моделей.

При удалении строк из таблицы, срабатывает экшен actionDel контроллера SiteController. Он обращается к методу delFromSpecs модели Specs. Данный метод позволяет удалить из сессии не нужные более данные.

Нижний блок, предназначенный для добавления моделей из списка, представляет собой форму с выпадающем списком. В эту форму передаётся массив с названиями всех моделей телевизионных приставок, записанных в БД. При нажатии кнопки «Добавить ТВбокс», происходит обращение к методу actionAdd контроллера SiteController и происходит еще одна итерация с использованием сессии.

**3.3.4** Разработка страницы «Скидки и распродажи».

Страница скидок представляет собой небольшие записи, содержащие минимально необходимую информацию:

* изображение и наименование телевизионной приставки;
* краткое описание акции и ее сроки;
* информацию о цене и скидочный купон;
* ссылка на страницу акции в магазине партнёре.

Для целей улучшения зрительного восприятия и придания некой визуальной ритмичности, записи располагаются в шахматном порядке.

На рисунке 3.8 изображена страница с информацией о скидках и распродажах.

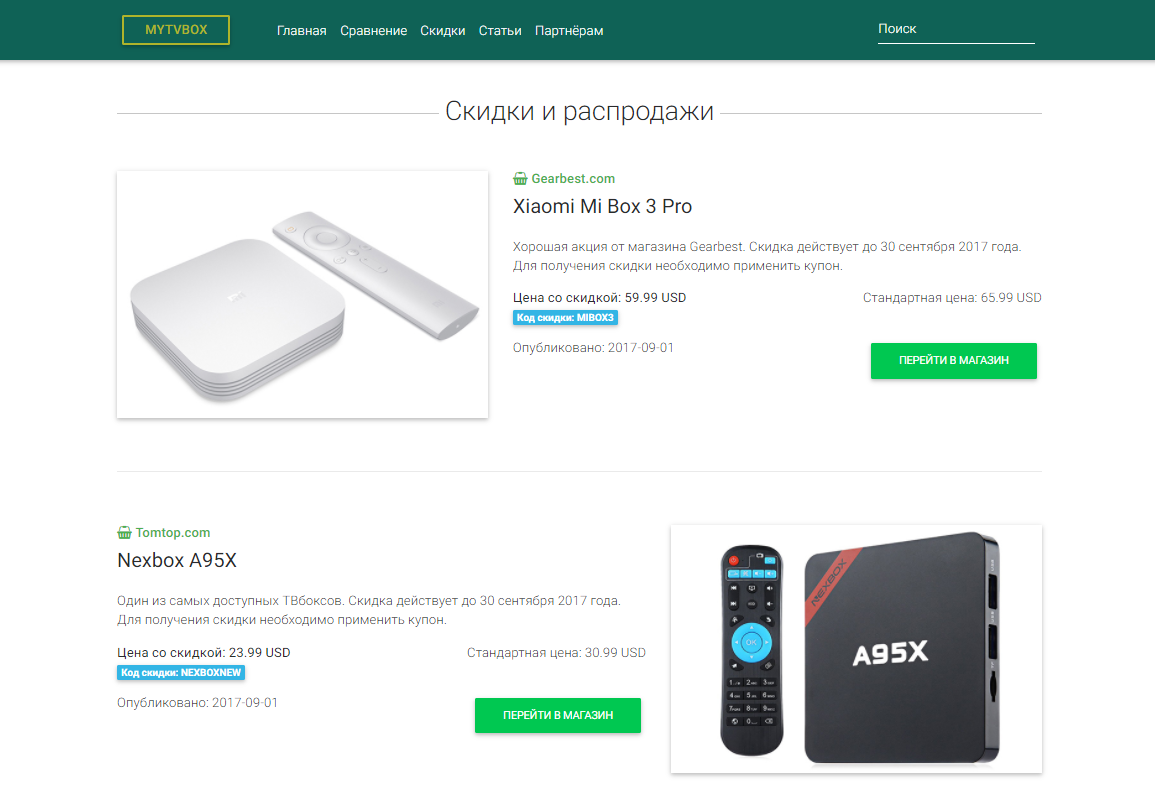


Рисунок 3.8 – Страница «Скидки и распродажи»

Контроллером данной страницы является файл SaleController.php. У данного класса всего один метод actionSale.

Данный метод обращается к моделям Shop и Sale и получает информацию из соответствующих таблиц БД. Далее полученная информация в виде ассоциативного массива передаётся в представление discount.php.

В представлении, с помощью цикла foreach, динамически генерируется необходимое количество записей.

**3.3.5** Разработка страницы «Миниатюры статей».

На рисунке 3.9 изображена страница, на которую выводится превью информационных статей. Название каждой статьи сопровождается картинкой и информацией о дате публикации.

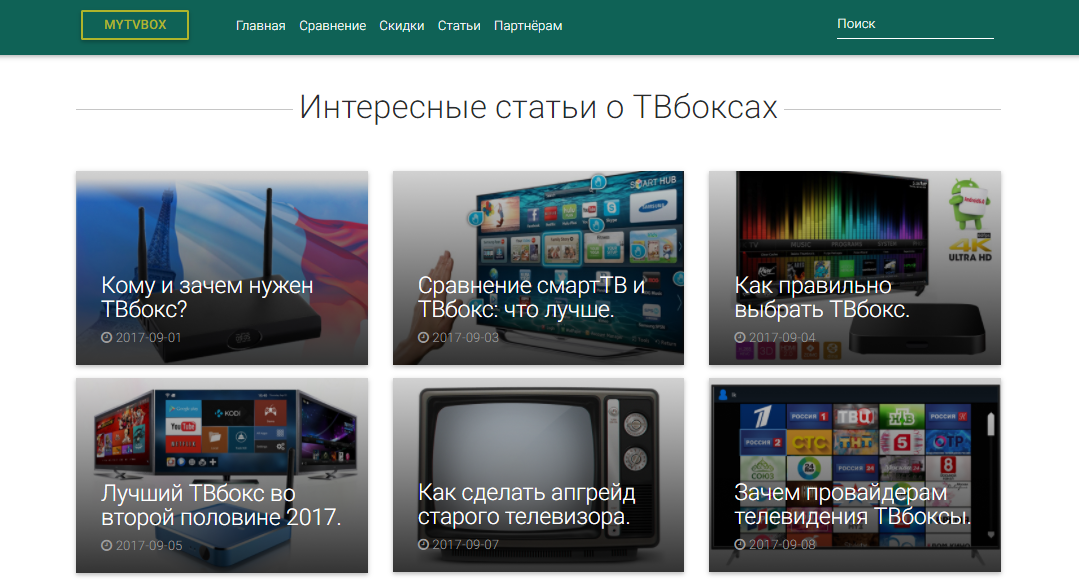


Рисунок 3.9 – Страница «Миниатюры статей»

Контроллером данной страницы является ArticlesController. Метод actionSeoarticles обращается к модели Article и получает необходимую информацию из таблицы article. После этого, с помощью метода render в представление seo.php передается ассоциативный массив с данными из БД. Далее с помощью цикла foreach динамически генерируется необходимое количество миниатюр статей.

* + 1. Разработка страницы «Информационная статья».

На рисунке 3.10 изображена страница информационной статьи.

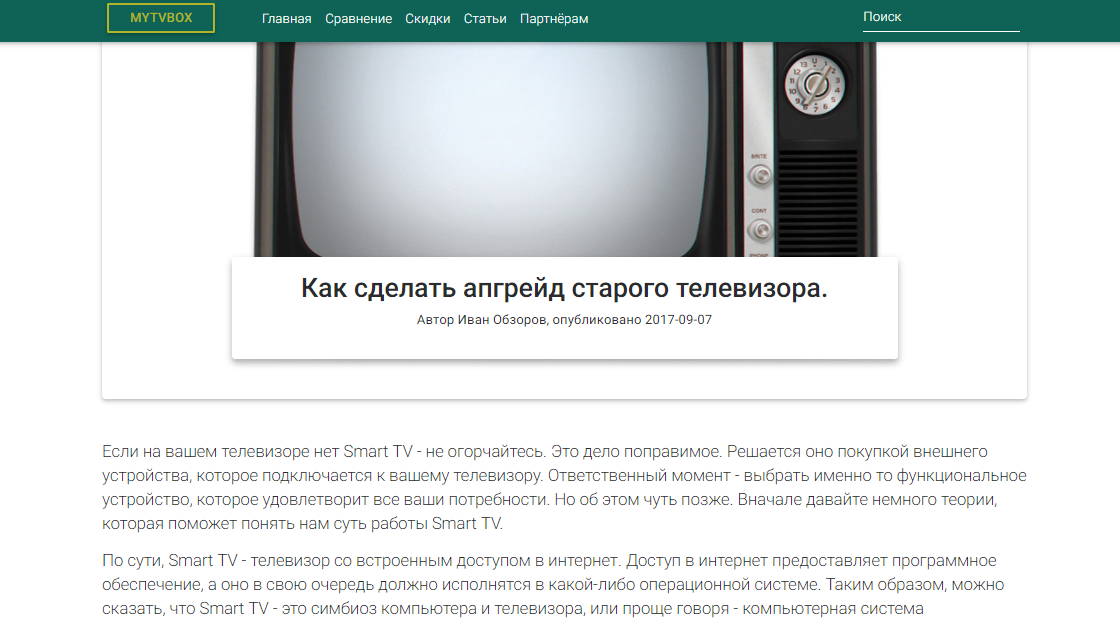


Рисунок 3.10 – Страница «Статья»

Логически страницу можно поделить на три блока:

* название статьи, сопровождающее изображение, информация об авторе и дата публикации;
* тело статьи;
* слайдер со ссылками на другие статьи.

Контроллером данной страницы также является ArticlesController. В связи с тем, что работа ведется с той же моделью, что и на странице миниатюр, мной было принято решение не создавать новый контроллер, а воспользоваться существующим.

С помощью метода actionQuestion и модели Article, из базы данных извлекается информация в соответствии с id параметром страницы. После этого с помощью метода render данные передаются в представление seo.php, где в свою очередь подставляются в шаблон страницы.

* + 1. [Разработка страницы](file:///H:\ТЗ-ИИТ\диплом.docx#_Toc289020721) с информацией для партнёров.

Страница с информацией для партнеров, изображённая на рисунке 3.11.

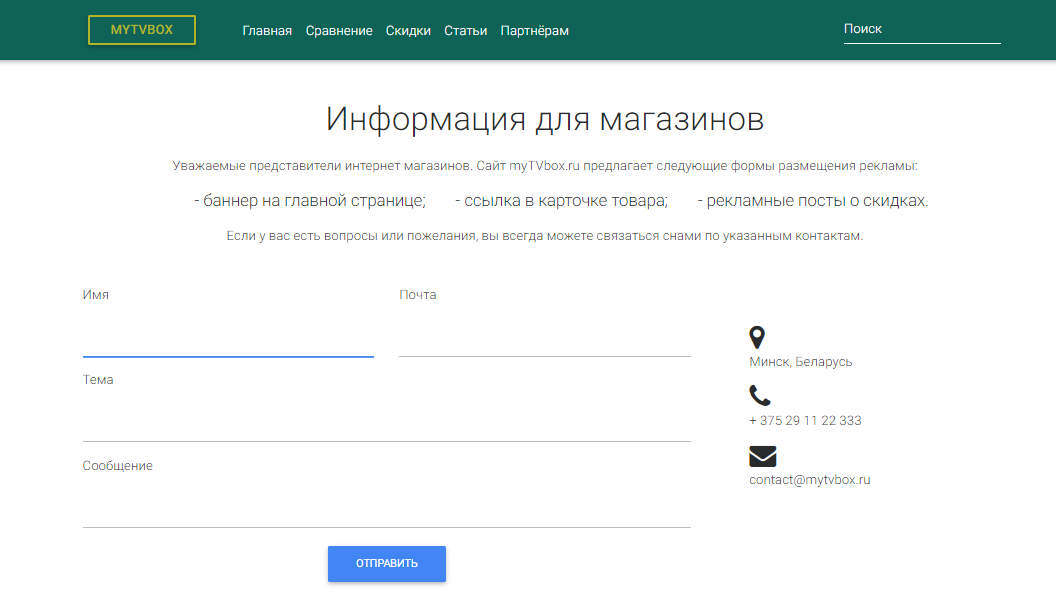


Рисунок 3.11 – Страница с информацией для партнёров

Логически страница состоит из трёх блоков:

* текстовая информация с вариантами сотрудничества;
* форма для отправки сообщений;
* контактная информация.

Построение страницы осуществляется контроллером contactController, методом actionContact. Данный метод создаёт новый экземпляр класса ContactForm.

В свою очередь модель ContactForm наследует класс Model и объявляет свойства name, email, subject и body. Эти свойства – поля формы на странице представления contact.php.

Помимо указанных выше свойств, модель ContactForm имеет два метода:

* rules – устанавливает обязательные для заполнения поля формы и производит валидацию поля email;
* contact – используя метод compose класса mailer, формирует электронное письмо и отправляет по указанному адресу.

В качестве параметра метод contact принимает адрес электронной почты, на который требуется отправить письмо.

Контроллер contactController получает из представления методом POST данные внесенный в форму пользователем и, в случае успешной валидации, записывает в сессию сообщение об успешной отправке письма. После этого обращается к представлению contact.php и с помощью метода render передает данные для генерации формы.

На странице представления или появляется сообщение об успешной отправке письма, или генерируется форма, в зависимости от того, как пользователь попал на страницу.

**3.3.8** [Разработка страницы](file:///H:\ТЗ-ИИТ\диплом.docx#_Toc289020721) управления контентом.

Построение страницы осуществляется контроллером adminController, методом actionEdit. Данный метод создаёт новый экземпляр класса CKEditor.

На странице представления появляется форма для авторизации. В случае ввода установленного имени и пароля, генерируется страница с доступом к нескольким различным модулям:

* модуль для добавления/редактирования/удаления страницы телевизионной приставки;
* модуль для добавления/редактирования/удаления новости о распродаже;
* модуль для добавления/редактирования/удаления новой информационной статьи;
* модуль для администрирования комментариев.

**4** **ТЕСТИРОВАНИЕ САЙТА**

**Тестирование сайта – важный этап в разработке любого программного продукта, включая веб сайты даже небольшой сложности.** Цель тестирования - это всесторонняя проверка ожидаемой работой системы и соответствие реальной производительности предъявляемым требованиям.

Под тестированием сайта понимается проверка соответствия заявленным характеристикам и требованиям, требованиям эксплуатации в различных окружениях, с различными нагрузками, требованиям по безопасности, требованиям по эргономике и удобству использования. В зависимости от направленности тестирования, проверяется та или иная особенность приложения или веб-сайта.

Для разработанного сайта были проведены следующие виды тестирования.

***Функциональное тестирование.*** Поскольку возможность пользоваться сайтом - основное требование, то проверка функциональности является первостепенной задачей. Выполнение целевых действий без нарушений с заранее определенным результатом, интеграция с другими программами и безопасность - только такая работа сайта может считаться эталонной. Функциональное тестирование проверяет соответствие реализованных функций требованиям технического задания [7].

**На рисунке 4.1 приведены результаты функционального тестирования.**

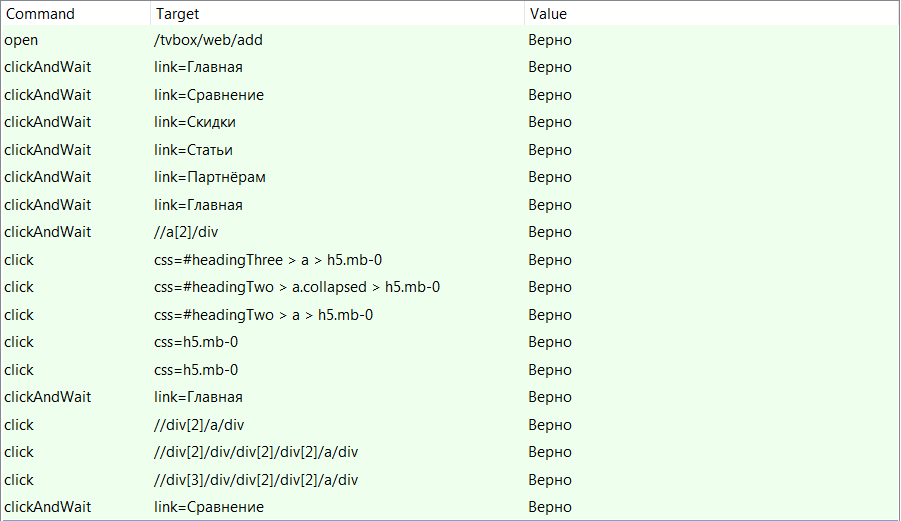


Рисунок 4.1 – Результат тестирования сайта

Тестирование проводилось при помощи плагина браузера Mozilla Firefox Selenium IDE (version 2.9.1). Была проверена работа каждой заявленной функции приложения. Также были исследованы возможные сценарии использования, протестированы ссылки, проверен статичный контент и динамические элементы.

***Конфигурационное тестирование.*** Этот вид тестирования позволяет проверить как приложение ведет себя при различных разрешениях экрана, в различных браузерах, на различных ОС, с разным программным и аппаратным обеспечением.

Разработанный сайт тестировался при помощи браузеров

* Microsoft Edge 40.15063;
* Mozilla Firefox 49.0 - 51.0;
* Opera 45.0 - 47.0;
* Google Chrome 59.0 - 61.0.

Дизайн сайта на всех браузерах отображается корректно, все функции работают правильно. Так же сайт протестирован при различных разрешениях экрана:

* ширина экрана 750 px;
* ширина экрана 970 px;
* ширина экрана 1170 px.

Сайт адаптируется под любой размер экрана, согласно стандарту библиотеки Bootstrap.

***Тестирование безопасности.*** Такой вид тестирования позволяет убедиться, что данные хранятся надежно, доступ к ним блокирован для посторонних лиц. Данные в процессе хранения, обработки и иной работы с ними не могут быть получены методами несанкционированного доступа. Проверяется защищенность БД, каналов связи, интерфейсов ввода и транспорта данных.

В ходе тестирования сайт проверен на возможные SQL, HTML и XSS инъекции. Установлено, что все веб-формы защищены и вводимы пользователем данные проходят валидацию перед добавлением в базу данных.

***Юзабилити тестирование сайта.*** Тестирование удобства пользования (юзабилити) – это вид тестирования, который проверяет удобство сайта и практичность в использовании.

Основные цели проведенного тестирования:

* выяснить насколько сайт понятен для окружающих;
* проверить удобство навигации;
* узнать, какое общее впечатление у пользователя;
* узнать, что на сайте лишнее и ненужное.

Для проведения юзабилити тестирования была создана фокус-группа из трёх человек:

* Пекарский Александр – специалист по BigData;
* Высоцкая Дарья – специалист по UI/UX;
* Высоцкая Светлана - продвинутый пользователь, знакомый с веб-технологиями и основами дизайна.

В процессе тестирования установлено, что сайт понятен пользователю с первых секунд посещения. Проблем с навигацией нет, меню простое и информативное. Сайт вызывает ассоциации с современными технологиями, отвечает ожиданиям пользователей и соответствует своей тематике. Лишних элементов навигации и дизайна не выявлено. Все используемые иконки хорошо читаются и являются однозначными к контексту.

1. **ПРОДВИЖЕНИЕ САЙТА**

**5.1 SEO продвижение**

Для составления эффективной стратегии поискового продвижения проекта в поисковых системах необходимо провести анализ сайтов конкурентов [8]. Анализ сайтов конкурентов дал следующие результаты:

**Заголовок страницы:**

* *INVIN (https://smartpristavka.ru)* - Главная- Андроид смарт ТВ приставка INVIN. Smartpristavka.ru - Длина 60 символов;
* *ТВ-приставка (https://android-tv-pristavka.ru)* – ТВ Приставки на Андроид - SMART TV android приставки для телевизоров с бесплатными каналами - Длина 91 символ;
* *Покупка5 (http://pokupka-5.ru)* **-** Купить Интернет ТВ приставки. Смарт ТВ приставки, Андроид ТВ приставки. Smart TV Android медиаплееры. Windows мини ПК. Tronsmart, HiMedia Q10 Pro, Zidoo X9S, MXQ Pro, X92, Egreat A5, Mecool M8S Pro, Vorke Z1, Z3, Vorke V1, V2, H96 Pro Plus, Tanix TX8 - Длина 255 символов;
* *DUNE HD (http://www.dune.ru)* – Dune HD – универсальный сетевой медиаплеер – Длина 42 символа;
* *Android TVBox и MiniPC (http://hd-mediaplayers.ru)* **-** Android TV Box & Mini PC — HD-mediaplayers.ru – Длина 45 символов;

**Описание страницы:**

* *INVIN (https://smartpristavka.ru)* **-** Приставки Smart TV на Android . Все в наличии! Гарантия 1 год! Оплата при получении! Доставка по России! Длина 104 символа;
* *ТВ-приставка* ***(****https://android-tv-pristavka.ru****)* –** Ищете ТВ приставку на Android? Купите Smart TV приставки по цене от 3000 рублей в Интернет Магазине в Москве Гарантия качества Доставка в регионы России – Длина 155 символов;
* *Покупка5* ***(****http://pokupka-5.ru****)*** *-* Купить Интернет ТВ приставки. Смарт ТВ приставки, Андроид ТВ. Android медиаплееры. Windows мини ПК. HiMedia Q10 Pro, Zidoo X9S, MXQ Pro, X92, Egreat A5, Mecool, Vorke Z1, Z3, Vorke V1, H96 Pro, Tanix-Длина 199 символов;
* *DUNE HD (http://www.dune.ru)* – Медиаплееры Dune HD в России от производителя – Длина 46 символов;
* *Android TVBox и MiniPC (http://hd-mediaplayers.ru)* **–** описание отсутствует.

**Заголовки h1-h3:**

* *INVIN* *(https://smartpristavka.ru)* – [h1] Главная- Андроид смарт ТВ приставка INVIN. Smartpristavka.ru, [h1] &nbsp, [h2] &nbsp, [h3] Информация, [h3] Сертификат соответствия;
* *ТВ-приставка (https://android-tv-pristavka.ru)* **–** [h2] ТВ приставка с Android сделает из любого телевизора SMART TV!, [h3] X96 TV Box Приставка на Android 6 (2 ГБ ОЗУ), [h3] MXQ Pro 4K TV Box Приставка на Android 5 (1 ГБ ОЗУ), [h3] NEXBOX A95X Smart TV Box Android 6 S905X ТВ Приставка, [h3] V88 TV Box Приставка на Android 5 (1 ГБ ОЗУ);
* *Покупка5 (http://pokupka-5.ru)* – [h1] Наши преимущества, [h3] Наши новости! Подпишись!, [h3] Отзывы и комментарии, [h3] Поиск по каталогу, [h3] Каталог;
* *DUNE HD (http://www.dune.ru) –* [h2] Обратный звонок, [h2] Покупка в рассрочку без переплаты, [h2] Это проще простого, [h2] Просто и удобно, [h3] Настоящему мужчине, [h3] Прекрасной даме, [h3] Детям и подросткам;
* *Android TVBox и MiniPC (http://hd-mediaplayers.ru)*-[h1] Android TV Box &amp Mini PC, [h1] Popcorn Hour RockBox Basic: ТВ бокс на основе Rockchip RK3328 с 1/8 Гб и Android 7.1.2, [h1] NAS Synology DiskStation DS418play рассчитан на использование дома, [h1] MSI Vortex G25: новый мини-ПК в консольном форм-факторе, [h1] Amazon Fire TV 2017: новый ТВ бокс на базе Amlogic S905L с поддержкой 4K HDR-10, [h1] Гибридные DVB TV приставки MECOOL KI-II-III PRO, [h1] Распродажа ТВ Бокса Mecool M8S PRO W.

**Поисковые фразы и видимость в Яндекс и Google.**

Видимость сайта– это показатель, определяющий количество показов по выбранным запросам тематической рубрики, исходя из занимаемых позиции сайта в результатах выдачи поисковой системы.

Проверка видимости сайта в контексте ключевых слов позволяет анализировать самые популярные страницы, запросы по которым ресурс занял лидирующие позиции в поисковых системах, упоминания в социальных сетях, а также видимость конкурентов и соотношение пересечения их показов с анализируемым сайтом.

Для сайта ТВ-приставка (https://android-tv-pristavka.ru) ключевых слов, по которым пользователи переходят на сайт из поисковых систем Яндекс и Google, не обнаружено.

В таблице 5.1 отражены ключевые слова, по которым пользователи наиболее часто переходят на сайт INVIN (https://smartpristavka.ru) из поисковых систем.

Таблица 5.1 – Позиции сайта INVIN в поисковых системах Яндекс и Google

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Позиции в Яндекс | | | Позиции в Google | | |
| Ключевое слово | Позиции | Показов в месяц | Ключевое слово | Позиции | Показов в месяц |
| Приставки | 72 | 331 | Как перепрошить андроид | 18 | 644 |
| Тв приставка | 30 | 268 | Приставка | 34 | 588 |
| Приставка смарт тв | 4 | 197 | Как прошить андроид | 27 | 487 |
| Тв-приставка | 29 | 165 | Приставки | 61 | 331 |
| Смарт тв приставка | 6 | 163 | Тв приставка | 16 | 268 |

В таблице 5.2 отражены ключевые слова, по которым пользователи наиболее часто переходят на сайт Покупка5 (http://pokupka-5.ru) из поисковых систем.

Таблица 5.2 – Позиции сайта Покупка5 в поисковых системах Яндекс и Google

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Позиции в Яндекс | | | Позиции в Google | | |
| Ключевое слово | Позиции | Показов в месяц | Ключевое слово | Позиции | Показов в месяц |
| Мини-пк | 46 | 348 | Приставки | 74 | 331 |
| Sj5000 | 49 | 131 | Chuwi-hi8 | 36 | 317 |
| Андроид тв приставка | 22 | 116 | Тв приставка | 46 | 268 |
| M8s | 11 | 98 | Тв тюнер для компьютера | 50 | 204 |

В таблице 5.3 отражены ключевые слова, по которым наиболее часто пользователи переходят на сайт *DUNE HD* (http://www.dune.ru) из поисковых систем.

Таблица 5.3 – Позиции сайта DUNE HD в поисковых системах Яндекс и Google

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Позиции в Яндекс | | | Позиции в Google | | |
| Ключевое слово | Позиции | Показов в месяц | Ключевое слово | Позиции | Показов в месяц |
| Kartina tv | 35 | 14,26 | Kartina tv | 14 | 285,2 |
| Картина тв | 32 | 12,62 | Твбокс | 18 | 252,4 |
| Твбокс | 39 | 1,58 | Kartina | 40 | 1,58 |
| Картина тв бесплатно | 37 | 0,13 | Картина бесплатно | 24 | 0,65 |

В таблице 5.4 отражены ключевые слова, по которым пользователи наиболее часто переходят на сайт Android TVBox и MiniPC (http://hd-mediaplayers.ru) из поисковых систем Яндекс и Google.

Таблица 5.4 – Позиции сайта Android TVBox и MiniPC в поисковых системах Яндекс и Google

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Позиции в Яндекс | | | Позиции в Google | | |
| Ключевое слово | Позиции | Показов в месяц | Ключевое слово | Позиции | Показов в месяц |
| Медиаплеер | 29 | 1020 | Медиаплееры | 33 | 210 |
| Мини пк | 22 | 348 | Fs videobox 4pda | 22 | 136 |
| Android tv box | 13 | 219 | Fs videobox скачать на андроид | 16 | 124 |
| Xiaomi box | 16 | 151 | Iconbit | 11 | 94 |

Учитывая вышеприведённые данные, был проведён анализ содержания заголовков (h1, h2, h3) сайтов конкурентов, использование ими ключевых слов, а также позиции в поисковых системах.

Наиболее популярные ключевые слова были выделены и при разработке сайта послужили основой для текстов заголовков (h1, h2, h3), были включены в мета-тэги title, description, keywords, а также стали основой для семантического ядра.

Семантическое ядро – это группа целевых запросов, которые вводят пользователи в строки поиска Яндекс, Рамблер, Google, Yahoo и других поисковых систем. Эти ключевые запросы должны отражать тематику сайта, т.е. отвечать на вопрос «о чем этот сайт?».

Для наиболее быстрого вхождения в поисковую выдачу Яндекс и Google, были написаны продвигающие статьи с использованием следующих ключевых слов:

* приставки;
* приставка смарт тв;
* андроид тв приставка;
* медиаплеер;
* твбокс;
* IPTV;
* xiaomi mi box.
  1. **Разработка новых функциональных возможностей**

Дальнейшее развитие сайта предполагает в первую очередь наращивание важного и полезного для пользователей контента. Помимо добавления страниц с информацией о новых моделях телевизионных приставок, будет расширено семантическое ядро сайта и написаны дополнительные информационные статьи.

По мере увеличения числа пользователей, также будет расширен функционал сайта. Одно из перспективных направлений существующего сегодня e-commerce бизнеса - это внедрение сервисов по возврату части стоимости товара после покупки, так называемый кэшбек.

Кешбэк (от англ. c*ashback –* возврат наличных денег) - термин, который используется в сферах интернет торговли, банковского дела и игорного бизнеса в качестве обозначения разновидности бонусной программы для привлечения клиентов и повышения их лояльности. В розничной и интернет торговле кешбэк - это отсроченная скидка, возвращаемая клиенту по истечении периода, отведённого на обмен/возврат [9].

Для внедрения подобного сервиса предстоит провести следующую работу:

* разработать и внедрить личный кабинет пользователя;
* подключится к API партнерских программам CPA сетей;
* подключится к API платежных систем.

Второе важное и перспективное направление развития сайта – переход к такой модели наполнения сайта контентом, когда его добавляют именно сами пользователи.

Такой подход хорошо подойдет для страницы сайта, на которой публикуются текущие скидки и распродажи. Для воплощения этого модуля потребуется:

* регистрация пользователя и личный кабинет;
* ограниченный доступ пользователей, публикующих материал, к базе данных и панели администратора (публикация и редактирование только своего материала).

Третье запланированное важное направление развития сайта – это интеграция с сервисами Яндекс, а именно в части импортирования отзывов пользователей из Яндекс Маркет. Для этого потребуется одобрение со стороны администрации Яндекс и подключение к API Маркета.

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения дипломного проекта была достигнута основная цель работы - разработан сайт телевизионных приставок.

Для достижения поставленной цели была выявлена основная целевая аудитория и был проведён анализ сайтов конкурентов. Это позволило правильно спланировать функциональные требования проекта, а также избежать ряда ошибок, допущенных разработчиками сайтов конкурентов.

В разработанном сайте телевизионных приставок удачно сочетается удобный, привлекательный интерфейс с функциональностью и простотой использования.

Разработана удобная и доступная схема подачи контента пользователю. Информация о телевизионных приставках, размещенная на главной странице, а также в карточках товаров удачно сочетает графический и структурный стиль.

Созданная структура базы данных позволяет без проблем масштабировать проект и добавлять любое необходимое количество контента. Использование паттерна MVC в логике работы сайта допускает добавление новых функциональных возможностей.

По окончанию основного этапа разработки, в проекте были реализованы: карточка товара, страница сравнения технических характеристик, страница скидок и распродаж, блок информационных статей, а также были сформулированы и изложены на отдельной странице предложения о сотрудничестве для магазинов партнеров.

При разработке архитектуры, сайт был разделен на две части: клиентскую и администраторскую. Клиентская часть содержит те страницы, которые доступны для просмотра любому пользователю. Администраторская часть содержит инструменты управления сайтом, предоставляет удобный интерфейс для настройки клиентской части ресурса и работы с основной базой данных.

Разработанный сайт соответствует всем поставленным требованиям. Цель создать информативный, привлекательный и удобный для пользователей сайт достигнута.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Spark. Как определить свою ЦА: 4 совета + примеры кейсов [электронный ресурс].-2016. Режим доступа: https://spark.ru/startup/verstal/blog/14871/kak-opredelit-svoyu-tsa-4-soveta-primeri-kejsov –Дата доступа: 09.08.2017.

[2] Межрегиональная эргономическая ассоциация. Что такое эргономика? [электронный ресурс].-2015- Режим доступа: http://www.ergo-org.ru/ergo.html –Дата доступа: 10.08.2017.

[3] Хабрахабр. Руководство по проектированию реляционных баз данных [электронный ресурс].-2013- Режим доступа: https://habrahabr.ru/post/193136/ –Дата доступа: 12.08.2017.

[4] От Новичка До Профессионала. 6 нормальных форм БД: [электронный ресурс].-2015- Режим доступа: http://i-novice.net/6-normalnyx-form-bd/ –Дата доступа: 15.08.2017.

[5] Хабрахабр. MVC для веб: проще некуда. [электронный ресурс].-2015- Режим доступа: https://habrahabr.ru/post/181772/ –Дата доступа: 26.08.2017.

[6] Yiiframework. Полное руководство пользователя [электронный ресурс]. -2015.- Режим доступа: <https://yiiframework.com.ua/ru/doc/guide/2/start-workflow/> –Дата доступа: 26.08.2017.

[7] Протестинг. Функциональное тестирование или Functional Testing [электронный ресурс] -2016- Режим доступа: http://www.protesting.ru/testing/types/functional.html –Дата доступа: 16.09.2017.

[8] Pr-Cy. Анализ конкурентов и конкуренции [электронный ресурс]. -2016- Режим доступа: <http://pr-cy.ru/lib/seo/Analiz-konkurentov-i-konkurentsii> –Дата доступа: 18.09.2017.

[9] Википедия. Кэшбек [электронный ресурс]. - 2016.- Режим доступа: https://ru.wikipedia.org/wiki/%D0%9A%D0%B5%D1%88%D0%B1%D1%8D%D0%BA –Дата доступа: 18.09.2017.

**ПРИЛОЖЕНИЕ А**

**(обязательное)**

**Листинг кода**

**Исходный код контроллера siteController**

**class** SiteController **extends** AppController  
{  
 **public function** actionIndex()  
 {  
 $cards = Tvbox::*find*()  
 ->joinWith('chipset')  
 ->all();  
 $this->setMeta('myTvbox | Главная');  
 **return** $this->render('index', compact('cards'));  
 }

**public function** actionBox(){  
 $id = Yii::*$app*->request->get('id');  
 $err = Tvbox::*findOne*($id);  
 **if**(**empty**($err))  
 **throw new** HttpException(404, 'Такого ТВбокса нет');  
 $media = Tvbox::*find*()  
 ->with('specification', 'chipset', 'partners', 'shop', 'slider')  
 ->where(['id' => $id])  
 ->limit(1)  
 ->one();  
 $revs= Review::*find*()  
 ->where(['id\_tvbox' => $id])  
 ->all();  
 $this->setMeta('myTvbox | ' . $media->name, 'твбокс ' . $media->name, 'tvbox ' . $media->name);  
 $model = **new** ReviewForm();  
 **if** ($model->load(Yii::*$app*->request->post()) ) {  
 $model->id\_tvbox = Yii::*$app*->request->get('id');  
 $model->image = UploadedFile::*getInstance*($model, 'image');  
 **if**($model->image) $model->upload('img/', $model->image);  
 **else** $model->image = 'Anonymus.jpg';  
 **if**( $model->save() ) {  
 Yii::*$app*->session->setFlash('reviewFormSubmitted');  
 **return** $this->refresh();  
 }**else**{  
 Yii::*$app*->session->setFlash('error', 'Ошибка');  
 }  
 }  
 **return** $this->render('box', compact('media', 'revs', 'model'));  
 }

**public function** actionSearch(){  
 $search = Yii::*$app*->request->get('search');  
 $query = Tvbox::*find*()  
 ->where(['like', 'name', $search])  
 ->joinWith('chipset')  
 ->all();  
 $this->setMeta('myTvbox | Поиск');  
 **return** $this->render('search', compact('query', 'search'));  
 }

**public function** actionAdd(){  
 $this->setMeta('myTvbox | Сравнение');  
 $id = Yii::*$app*->request->get('id');  
 $box = Tvbox::*find*()->with('specification', 'chipset')->where(['id' => $id])->one();  
 $session=Yii::*$app*->session;  
 $session->open();  
 $specs = **new** Specs();  
 $specs->addToSpecs($box);  
 $it = Tvbox::*find*()->all();  
 **return** $this->render('compare', compact('session', 'it'));  
 }

**public function** actionDel(){  
 $id = Yii::*$app*->request->get('id');  
 $session=Yii::*$app*->session;  
 $session->open();  
 $specs = **new** Specs();  
 $specs->delFromSpecs($id);  
 $it = Tvbox::*find*()->all();  
 **return** $this->render('compare', compact('session', 'it'));  
 }  
}

**Исходный код модели ReviewForm**

**class** ReviewForm **extends** ActiveRecord  
{  
 **public static function** tableName()  
 {  
 **return** 'review';  
 }  
  
 **public function** attributeLabels()  
 {  
 **return** [  
 'id\_tvbox' => 'id',  
 'name' => 'Имя',  
 'email' => 'Почта',  
 'message' => 'Сообщение',  
 'image' => 'Фото для аватарки: ',  
 ];  
 }  
  
 **public function** rules()  
 {  
 **return** [  
 [['name', 'email', 'message', 'id\_tvbox'], 'required'],  
 ['email', 'email'],  
 ['image', 'trim'],  
 ];  
 }  
  
 **public function** upload($dir, $imageObj) {  
 $name = $imageObj->baseName;  
 $ext = $imageObj->extension;  
 $imageObj->saveAs($dir . $name . '.' . $ext);  
 **return true**;  
 }  
}

**Исходный код представления главной страницы**

<main>  
 <!--Main layout-->  
 <div class="container">  
 <!--Page heading-->  
 <div class="row wow fadeIn" data-wow-delay="0.2s">  
 <div class="col-md-12">  
 <div class="jumbotron">  
 <h2 class="h2-responsive">Хотите сделать старый телевизор умным?</h2>  
 <h3 class="title font-bold">Тогда вам нужен твбокс.</h3>  
 <br>  
 <p>  
 <i class="fa fa-gamepad fa-5x fa-fw" aria-hidden="true"></i>  
 <i class="fa fa-youtube fa-5x fa-fw" aria-hidden="true"></i>  
 <i class="fa fa-film fa-5x fa-fw" aria-hidden="true"></i>  
 <i class="fa fa-music fa-5x fa-fw" aria-hidden="true"></i>  
 <i class="fa fa-internet-explorer fa-5x fa-fw" aria-hidden="true"></i>  
 </p>  
 </div>  
 </div>  
 </div>  
 <!--/.Page heading-->  
 <!--First row-->  
 <div class="row mt-5 wow">  
 **<?php foreach** ($cards **as** $card): **?>** <!--First column-->  
 <div class="col-lg-4 wow fadeIn" data-wow-delay="0.2s">  
 <!--Card Dark 1-->  
 <div class="card">  
 <!--Card image-->  
 <div class="view overlay hm-white-slight">  
 **<?=** Html::*img*("@web/img/{$card->image}", ['alt' => $card->name, 'class' => 'img-fluid']) **?>** <a href="**<?=**Url::*to*(['site/box', 'id' => $card->id]) **?>**">  
 <div class="mask"></div>  
 </a>  
 </div>  
 <!--/.Card image-->  
 <!--Card content-->  
 <div class="card-body">  
 <!--Title-->  
 <h4 class="card-title">**<?=** $card->name**?>**</h4>  
 <hr>  
 <!--Text-->  
 <p class="card-text">**<?=** $card->snippet**?>**</p>  
 <div class="row">  
 <div class="col-7">  
 <div><i class="fa fa-microchip fa-1x fa-fw" aria-hidden="true"></i> **<?=** $card->chipset->cpu**?>**</div>  
 <div><i class="fa fa-database fa-1x fa-fw" aria-hidden="true"></i> **<?=** $card->specification->ram**?>** RAM</div>  
 </div>  
 <div class="col-5">  
 <div><i class="fa fa-hdd-o fa-1x fa-fw" aria-hidden="true"></i> **<?=** $card->specification->rom**?>** ROM</div>  
 <div><i class="fa fa-android fa-1x fa-fw" aria-hidden="true"></i> **<?=** $card->specification->android**?>** vers.</div>  
 </div>  
 </div>  
 <div class="row">  
 <a href="**<?=**Url::*to*(['site/add', 'id' => $card->id]) **?>**" data-id="**<?=**$card->id**?>**" class="col flex-row-reverse add-to-specs">  
 <div class="mbottom waves-effect waves-light p-2 ">Сравнить <i class="fa fa-plus"></i></div>  
 </a>  
 <a href="**<?=**Url::*to*(['site/box', 'id' => $card->id]) **?>**" class="col d-flex flex-row-reverse">  
 <div class="mbottom waves-effect waves-light p-2">Подробнее <i class="fa fa-chevron-right"></i></div>  
 </a>  
 </div>  
 </div>  
 <!--/.Card content-->  
 </div>  
 <!--/.Card Dark-->  
 </div>  
 <!--/.First column-->  
 **<?php endforeach**;**?>** </div>  
 <!--/.First row-->  
 </div>  
 <!--/.Main layout-->  
</main>

**Исходный код модели Specs**

**class** Specs **extends** ActiveRecord  
{  
 **public function** addToSpecs($box){  
 $\_SESSION['specs'][$box->id] = [  
 'name' => $box->name,  
 'image' => $box->image,  
 'ram' => $box->specification->ram,  
 'rom' => $box->specification->rom,  
 'wifi' => $box->specification->wifi,  
 'bluetooth' => $box->specification->bluetooth,  
 'android' => $box->specification->android,  
 'cpu' => $box->chipset->cpu,  
 'gpu' => $box->chipset->gpu,  
 'codec' => $box->chipset->codec,  
 'core' => $box->chipset->core  
 ];  
 }  
  
 **public function** delFromSpecs($id){  
 **unset**($\_SESSION['specs'][$id]);  
 }  
}

**Исходный код представления compare**

<main id="mainContainer">  
 <!-- Main Container -->  
 <div class="container">  
 <section class="section">  
 <!-- Shopping Cart table -->  
 <div class="table-responsive product-table table-hover">  
 <table class="table mb-5">  
 <!-- Table head -->  
 <thead>  
 <tr>  
 <th>Фото</th>  
 <th>Модель</th>  
 <th>Процессор</th>  
 <th>Графика</th>  
 <th>Память</th>  
 <th>Накопитель</th>  
 <th>Интерфейсы</th>  
 <th>Кодек</th>  
 <th>Android</th>  
 <th>Удалить</th>  
 </tr>  
 </thead>  
 <!--/.Table head -->  
 <!-- Table body -->  
 <tbody>  
 **<?php foreach**($session['specs'] **as** $id => $spec): **?>  
 <?php if**($id>0):**?>** <!-- First row -->  
 <tr>  
 <th scope="row">  
 **<?=** Html::*img*("@web/img/{$spec['image']}", ['alt' => $spec->name, 'class' => 'img-fluid z-depth-0']) **?>** </th>  
 <td>  
 <h5><strong>**<?=** $spec['name']**?>**</strong></h5>  
 </td>  
 <td>**<?=** $spec['core']**?> <?=** $spec['cpu']**?>**</td>  
 <td>**<?=** $spec['gpu']**?>**</td>  
 <td>**<?=** $spec['rom']**?>**</td>  
 <td>**<?=** $spec['ram']**?>**</td>  
 <td>**<?=** $spec['wifi']**?>**, **<?=** $spec['bluetooth']**?>**</td>  
 <td>**<?=** $spec['codec']**?>**</td>  
 <td>**<?=** $spec['android']**?>**</td>  
 <td>  
 <a href="**<?=**Url::*to*(['site/del', 'id' => $id]) **?>**" class="btn btn-sm btn-primary text-danger del-item" title="Удалить">X  
 </a>  
 </td>  
 </tr>  
 **<?php endif**;**?>** <!-- /.First row -->  
 **<?php endforeach**;**?>** </tbody>  
 </table>  
<!-- <div class="row">-->  
<!-- <div class="col-12 text-center">-->  
<!-- <button type="button" class="btn btn-primary"><i class="fa fa-plus left"></i> Добавить модель </button>-->  
<!-- </div>-->  
<!-- </div>-->  
 </div>  
 <!-- /.Shopping Cart table -->  
 <div class="row mb-5">  
 <div class ="col text-center">  
 **<?php** $items = ArrayHelper::*map*($it,'id','name');**?>  
 <?php** $param = ['class' => 'form-control-lg'];**?>  
 <?php if**(Url::*to*() == '/tvbox/web/site/add?id=' . Yii::*$app*->request->get('id') ) $link = ''; **else** $link = 'site/add'**?>** <form method="GET" action="**<?=**$link;**?>**">  
 **<?=** Html::*dropDownList*('id', 'null', $items, $param);**?>  
 <?=** Html::*submitButton*('Добавить ТВбокс', ['class' => 'btn btn-primary mb-3']);**?>** </form>  
 </div>  
 </div>  
 </section>  
 </div>  
 <!-- /.Main Container -->  
</main>

**Исходный код модели ContactForm**

**class** ContactForm **extends** Model  
{  
 **public** $name;  
 **public** $email;  
 **public** $subject;  
 **public** $body;  
  
 **public function** rules()  
 {  
 **return** [  
 [['name', 'email', 'subject', 'body'], 'required'],  
 ['email', 'email'],  
 ];  
 }  
  
 **public function** contact($email)  
 {  
 **if** ($this->validate()) {  
 Yii::*$app*->mailer->compose()  
 ->setTo($email)  
 ->setFrom([$this->email => $this->name])  
 ->setSubject($this->subject)  
 ->setTextBody($this->body)  
 ->send();  
 **return** true;  
 }  
 **return** false;  
 }  
}

**Исходный код представления**

<main id="mainContainer">  
 <!-- Contact Form -->  
 <section class="section">  
 <div class="container">  
 <div class="row">  
 <div class="col-md-12">  
 <h1 class="text-center m-4">Информация для магазинов</h1>  
 <div class="text-center mb-3">  
 Уважаемые представители интернет магазинов. Сайт myTVbox.ru предлагает следующие формы размещения рекламы:  
 <ul class="mt-3">  
 <li class="d-inline m-3 lead"> - баннер на главной странице;</li>  
 <li class="d-inline m-3 lead"> - ссылка в карточке товара;</li>  
 <li class="d-inline m-3 lead"> - рекламные посты о скидках.</li>  
 </ul>  
 </div>  
 <div class="text-center mb-5">  
 Если у вас есть вопросы или пожелания, вы всегда можете связаться снами по указанным контактам.  
 </div>  
 </div>  
 </div>  
 **<?php if** (Yii::*$app*->session->hasFlash('contactFormSubmitted')): **?>** <div class="alert alert-success text-center h5 p-5">  
 Спасибо, что связались с нами. Мы вам ответим, как можно скорее.  
 </div>  
 **<?php else**: **?>** <div class="row">  
 <div class="col-md-8 mb-r">  
 **<?php** $form = ActiveForm::*begin*(['id' => 'contact-form']); **?>** <div class="row">  
 <!-- First column -->  
 <div class="col-md-6">  
 **<?=** $form->field($model, 'name')->textInput(['autofocus' => **true**])->label("Имя") **?>** </div>  
 <div class="col-md-6">  
 **<?=** $form->field($model, 'email')->label("Почта") **?>** </div>  
 </div>  
 <div class="row">  
 <div class="col-md-12">  
 **<?=** $form->field($model, 'subject')->label("Тема") **?>** </div>  
 </div>  
 <div class="row">  
 <!-- First column -->  
 <div class="col-md-12">  
 **<?=** $form->field($model, 'body')->label("Сообщение") **?>** </div>  
 </div>  
 <div class="form-group text-center">  
 **<?=** Html::*submitButton*('Отправить', ['class' => 'btn btn-primary', 'name' => 'contact-button']) **?>** </div>  
 **<?php** ActiveForm::*end*(); **?>** </div>  
 <!-- Second column -->  
 <div class="col-md-4 mt-5">  
 <ul class="contact-icons">  
 <li><i class="fa fa-map-marker fa-2x"></i>  
 <p>Минск, Беларусь</p>  
 </li>  
 <li><i class="fa fa-phone fa-2x"></i>  
 <p>+ 375 29 11 22 333</p>  
 </li>  
 <li><i class="fa fa-envelope fa-2x"></i>  
 <p>contact@mytvbox.ru</p>  
 </li>  
 </ul>  
 </div>  
 <!-- /.Second column -->  
 </div>  
 **<?php endif**; **?>** </div>  
 </section>  
 <!-- /.Contact Form -->  
</main>

**Исходный код файла main.js**

**new** WOW().init();  
  
$('.add-to-specs').on('click', **function**(e){  
 e.preventDefault();  
 $(**this**).addClass( "color-link" );  
 $('.color-change').addClass( "color-link" );  
 **var** id = $(**this**).data('id');  
 $.ajax({  
 url: 'site/add',  
 data: {id: id},  
 type: 'GET',  
 success: **function**(res){  
 **if**(!res) alert('Ошибка!');  
 console.log(res);  
 },  
 error: **function**(){  
 alert('Error!');  
 }  
 });  
});