МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования   
«Белорусский государственный университет транспорта»

Кафедра «Учетные системы и технологии бизнес-менеджмента»

Курсовая работа

по дисциплине «Современные технологии проектирования информационных систем»

на тему:

«Программное средство Web-приложение (интернет-магазин)»

Выполнил Проверил

студент группы ГИ-41 к.т.н., доц.

Зайцев И. С. Быченко О. В.

Гомель, 2022

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc121215669)

[1 АЛАНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 5](#_Toc121215670)

[2 РАЗРАБОТКА ТРЕБОВАНИЙ 7](#_Toc121215672)

[3 РАЗРАБОТКА АРХИТЕКТУРЫ WEB-ПРИЛОЖЕНИЯ 10](#_Toc121215673)

[4 ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА 13](#_Toc121215674)

[ЗАКЛЮЧЕНИЕ 23](#_Toc121215675)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 24](#_Toc121215676)

# ВВЕДЕНИЕ

Целью данной работы является написание требований к веб-сайту и создание прототипа интерфейса веб-сайта «Интернет-магазин продажи электронных товаров» по ранее написанным требованиям. При создании сайта, первоначально, будет проведен аналитический этап, по результатам которого станет известно, что делать и зачем. Далее будет рассмотрена архитектура web-приложения. Прототип веб-сайта будет создан при помощи популярных технологий создания веб-страниц, таких как HTML, CSS и JavaScript.

Результатом работы станет прототип веб-сайта «Интернет-магазин продажи электронных товаров».

# АЛАНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## Описание предметной области интернет-магазина

Интернет-магазин — сайт, торгующий товарами в интернете. Позволяет пользователям сформировать заказ на покупку, выбрать способ оплаты и доставки заказа в сети Интернет.

Выбрав необходимые товары или услуги, пользователь обычно имеет возможность тут же на сайте выбрать метод оплаты и доставки. Совокупность отобранных товаров, способ оплаты и доставки представляют собой законченный заказ, который оформляется на сайте путем сообщения минимально необходимой информации о покупателе. Информация о покупателе может храниться в базе данных магазина если бизнес-модель магазина рассчитана на повторные покупки, или же отправляться разово. В интернет-магазинах, рассчитанных на повторные покупки, также ведется отслеживание возвратов посетителя и история покупок. Часто при оформлении заказа предусматривается возможность сообщить некоторые дополнительные пожелания от покупателя продавцу.

Интернет-магазины создаются с применением систем управления контентом сайтов, оснащенных необходимыми модулями. Крупные интернет-магазины работают на специально для них разработанных или адаптированных типовых системах управления. Средние и малые магазины обычно используют типовое коммерческое и свободное ПО.

Система управления контентом сайта интернет-магазина может быть коробочным продуктом, самостоятельно устанавливаемым на [хостинг](https://infolisting.ru/text/category/hosting/)-площадку, может быть частной разработкой программистов, ими же обслуживаемой, или может быть программным сервисом, предоставляемым с помесячной оплатой.

Нужды администраторов интернет-магазина в складском, торговом, бухгалтерском и налоговом учете должны поддерживаться невидимой посетителям частью интернет-магазина — back-end’om. Экономически эффективной практикой создания интернет-магазинов является применение специализированных систем учета. Интернет-магазин обычно интегрирован с такими системами учета.

Есть две разновидности интернет-магазинов, в зависимости от вида торговли:

1) Магазины, которые продают товар со своего склада. Такой магазин – прекрасный вариант дополнительного сбыта товара, обычно дают более низкую цену, чем даже в своем реальном магазине;

2) Магазины, которые продают товар других магазинов/людей. Это может быть торговля внyтри страны, либо [международная торговля](https://infolisting.ru/text/category/mezhdunarodnaya_torgovlya/). В этом случае интернет-магазин зарабатывает на комиссии, которую платят продавцы за выставление товара. Здесь интернет-магазин выступает гарантом сделки между продавцом и покупателем. Такие магазины используют систему "репутация" продавца. Кроме того, покупатель может пожаловаться администрации сайта на продавца и получить необходимую помощь по возвращению денег, в случае обмана.

Также магазины могут отличаться по способу продажи:

1) Фиксированная цена товара - с доставкой, включенной в стоимость, либо с доставкой, которая считается отдельно, после оформления заказа (очень часто эффект низкой цены бывает испорчен из-за высокой стоимости доставки, продавцы нарочно могут ставить низкую стоимость на товар, а на доставку наоборот - высокую, на чем и зарабатывают).

2) Система аукциона - на товар объявляется аукцион. Кроме начальной цены, продавец может объявлять так называемую блиц-цену - это стоимость, за которую продавец готов отдать товар без торга. Есть такой нюанс, как скрытая цена - продавец ставит очень низкую цену на товар (чтобы при поиске товара, клиент заметил именно его лот), но включает опцию "минимальная ставка" - это [минимальная цена](https://infolisting.ru/text/category/minimalmznaya_tcena/), которая скрыта от глаз покупателя и он должен повышать ставки, пока не достигнет ее, иначе ставка не будет принята.

# РАЗРАБОТКА ТРЕБОВАНИЙ

1. Корректировка оптимизации

Для каждой страницы сайта должна быть возможность прописывать вручную уникальные title, description, keywords и h1.

При этом, нужно предусмотреть и автоматическую генерацию этих параметров, так как у интернет-магазина может быть несколько десятков тысяч страниц и вручную их все обработать нереально, да и бессмысленно. Достаточно задать шаблон, по которому все указанные теги будут генерироваться автоматически.

2. Возможность ручной настройки URL-адресов

Должна быть возможность не только прописывать произвольный URL, но и задавать нужную вложенность и принадлежность страницы к какому-либо разделу. При этом, сама система управления должна уметь правильно автоматически генерировать URL-адреса.

Например, если мы создаем дочернюю страницу в разделе «Телефоны», то мы могли бы получить вот такой адрес:

mysite.by/phones/apple

При этом, если у нас есть необходимость сделать URL-адрес другим, то функционально такая возможность должна быть предусмотрена.

3. Меню

На сайте должно быть два меню.

Первое меню — это главное меню, где находятся такие пункты как:

* Главная (со списком новых товаров (не более 8 штук));
* Все товары (с возможностью выбора способа отображения товаров (список, плитка));
* Контакты (вкладка должна содержать форму обратно связи, номера телефонов, почты и ссылки на сторонние сервисы для связи)
* О нас (содержит информацию об интернет-магазине, приветственный видеоролик и дополнительную информацию).

Второе меню — меню каталога товаров.

Оно должно располагаться сбоку и содержать следующие пункты:

* Категории;
* Производители;
* Фильтрацию по цене (выбор цены в определенных пределах);
* Сортировку товаров (по алфавиту, по цене, по дате загрузки);

В меню должна быть возможность выбора нескольких нужных пунктов.

4. Страница отдельного товара

Вверху страницы должны отображаться фотографии товара на который мы перешли. Заголовок должен быть под фотографиями товара. Далее отображаться цена в зависимости от выбранной валюты. Описание товара находиться немного ниже. Кнопки с возможностью добавления товара в корзину и список избранного.

На странице товара должна быть возможность выбрать нужную характеристику (Есть один товар, но разных цветов или, например, разного объема), чтобы избежать дублирование страниц.

Внизу страницы список с похожими товарами и возможность оставить комментарий (Если товаров на сайте много и товар не уникальный, то в 90% случаев и в описаниях товаров используется неуникальный контент. Пользовательский контент уникальный и полезный, кроме того, совершенно бесплатный).

Возможность множественной привязки одного товара к разным категориям.

Чтобы избежать дублей страниц, должна быть возможность задать товару один статичный URL и этот URL будет постоянным. Например, есть товар в категории “Телефоны”, имеет вот такой адрес: mysite.by/phones/xiaomi-p3-j12.

Предположим, что этот же товар, нужно показывать в другом разделе, например, «Xiaomi», расположенному по адресу: mysite.by/phones/xiaomi. Суть в том, что даже если товар будет показываться в разделе «Xiaomi», при клике на него, посетитель все равно должен попадать на страницу: mysite.by/phones/xiaomi-p3.

5. Страницы регистрации и входа

При регистрации пользователь должен обязательно заполнить три поля – логин и дважды пароль. Пароли должны совпадать.

На странице входа у пользователя должно отображаться несколько элементов: поля для ввода логина и пароля, кнопки “Забыл пароль” и “Запомнить меня”, и возможность входа через сторонние сервисы (Google, GitHub)

6. Корзина и список желаний

Должна присутствовать временные корзина и список избранного для гостей и постоянные для зарегистрированных пользователей.

В корзине отображается ранее добавленные в неё товары (фотография, название, цена, количество, общая стоимость). Должна быть предусмотрена возможность изменения количества товара в корзине и возможность удаления товара из корзины.

Содержит форму для заполнения адреса доставки (ФИО, страна, область, индекс, номер телефона) и кнопку для оформления заказа. Присутствует возможность выбора ранее сохранённого адреса и способа оплаты.

В списке желаний отображается ранее добавленные в него товары (фотография товаров, название, цена, количество, общая стоимость). Должна быть предусмотрена возможность изменения количества товара в корзине и возможность переноса товара из списка желаний в корзину.

7. Панель администратора

Должна содержать различные роли доступа и различный доступный функционал в зависимости от роли. Админ-панель должна содержать следующий функционал:

* Возможность добавления новых товаров
* Редактирование уже существующих товаров
* Возможность давать различные роли пользователя
* Возможность вечной и временной блокировки пользователей
* А также должна отображаться информация о пользователях и товарах

# РАЗРАБОТКА АРХИТЕКТУРЫ WEB-ПРИЛОЖЕНИЯ

В Django используется принцип DRY (don’t repeat yourself), то есть «не повторяй себя», что позволяет сократить время формирования. Это означает, что при использовании Django пользователю нет необходимости по несколько раз переписывать одни и те же коды, так как фреймворк предоставляет возможность создания сайтов из компонентов.

Django пригоден для проектирования веб-приложений, имеющих высокую загруженность. Это стало возможным благодаря использованию архитектуры фреймворка. Фреймворк Django был сформирован на языке программирования Python, поэтому его структурная организация соответствует особенностям этого языка. Проектировщики задействовали в Django шаблон MVC (Model-View-Controller, то есть, «Модель-Представление-Контроллер»), и он используется в текущем варианте фреймворка.

Архитектура MVC дает возможность разработчикам использовать визуальное представление и бизнес-логику приложения по отдельности. Следует отметить, что при работе с Django проектировщики более часто применяют термин MVT (Model-View-Template), то есть, модель-представление-шаблон. Компоненты MVT могут быть использованы независимо друг от друга. На рисунке ниже представлена схема архитектуры MVT в Django.

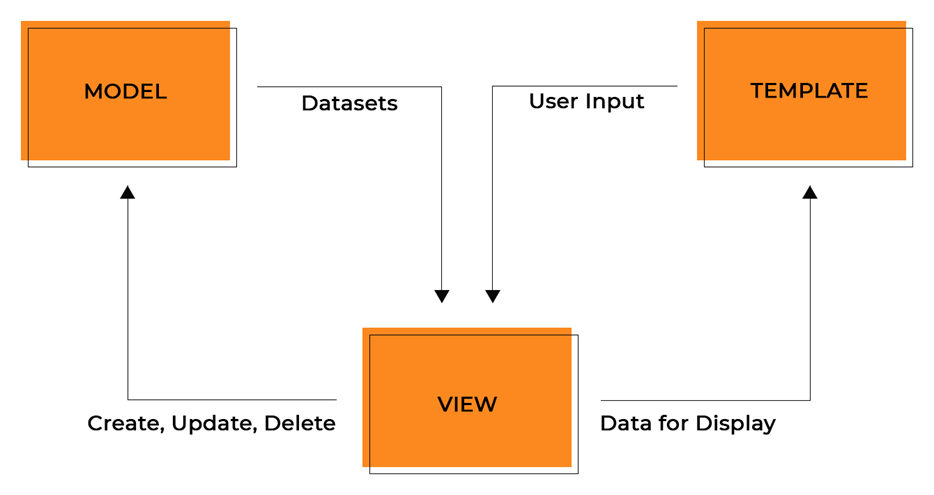


Рисунок 3.1 - Схема архитектуры MVT в Django

Документация Django определяет модель (model) в качестве источника информации о данных, в которых содержатся ключевые поля и поведение данных. Как правило, одна модель отображает одну таблицу в базе данных. Платформа Django способна поддерживать базы данных PostgreSQL, MySQL, SQLite и Oracle.

В моделях содержится информация о данных, которые представлены атрибутами или полями. Так как модель является простым классом, то ей ничего не известно о других уровнях Django. Взаимодействие между уровнями осуществляется через API. Модель определяет бизнес-логику, методы, свойства и другие компоненты, сопряженные с манипуляцией данными. Помимо этого, модели предоставляют разработчикам возможность создания, чтения, обновления и удаления объектов в базе данных.

Представление (view) способно решать следующие задачи:

* Прием HTTP-запросов.
* Реализация бизнес-логики, определенной методами и свойствами.
* Отправка HTTP-ответа в ответ на запросы.

Таким образом, представление должно получать данные от модели и предоставлять шаблонам (templates) доступ к этим данным или предварительно обрабатывать данные и затем предоставлять к ним доступ шаблонам. В Django имеется мощная поддержка шаблонов и собственный язык разметки. Шаблоны являются файлами с HTML-кодом, при помощи которого могут отображаться данные. Содержимое файлов может быть, как статическим, так и динамическим. Шаблоны не могут содержать бизнес-логику, поэтому они лишь отображают данные. На рисунке ниже изображен пример шаблона страницы.

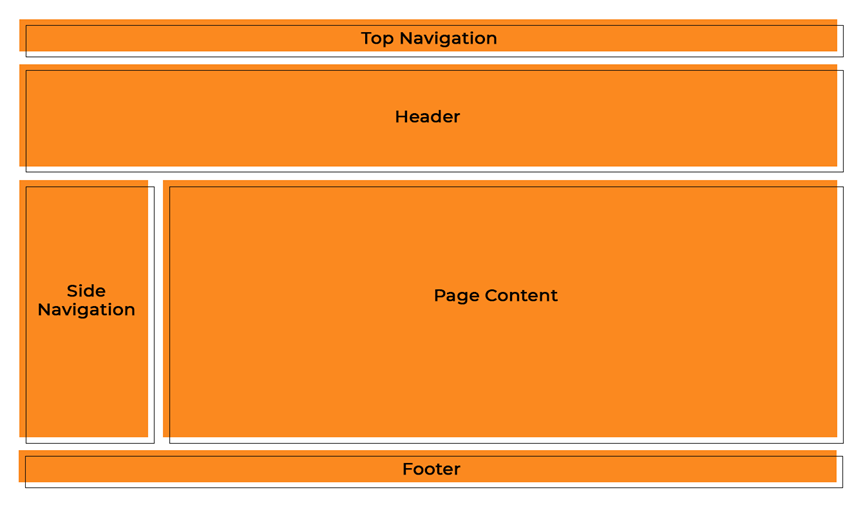


Рисунок 3.2 - Шаблон страницы

Данная архитектура предоставляет возможность Django успешно решать следующие задачи:

* Формирование движков для сайтов.
* Реализация систем CRM (Customer Relationship Management), то есть, систем управления отношениями с клиентами.
* Выполнение машинного обучения (machine learning).

Некоторые специалисты полагают, что Django является одной из многих систем управления контентом (CMS). Но в реальности Django является программным инструментом, при помощи которого обеспечивается возможность создания и запуска веб-приложения.

Программная платформа Django способна справиться с большим количеством задач и повышенными нагрузками. Ее используют для того, чтобы создать:

* Системы CRM.
* Системы CMS.
* Различные коммуникационные платформы.
* Сервисы бронирования номеров.
* Платформы управления документооборотом.

Помимо этого, Django может использоваться для формирования алгоритмических генераторов, платформ, предназначенных для электронных рассылок, систем верификации, систем фильтрации с динамическими правилами и сложными параметрами, платформ для анализа данных и сложных вычислений, машинного обучения.

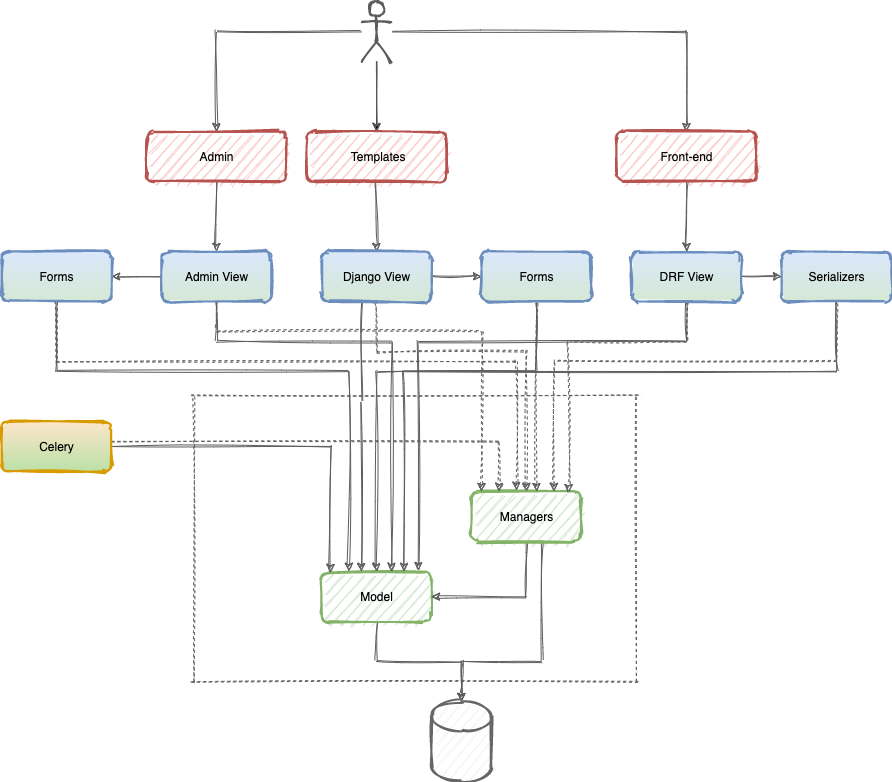


Рисунок 3.3 - Архитектура Web-приложения

# ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА

В жизни есть определенные правила, которые нельзя нарушать. В дизайне пользовательского интерфейса тоже есть правила, по которым нужно жить. Их называют «эвристикой» или общими принципами, улучшающими юзабилити интерфейсов. Это повторяемые паттерны, проверенные временем и помогающие пользователям ориентироваться в интерфейсе.

Далее рассмотрим спроектированный интерфейс веб-приложения “Интернет-магазин”:

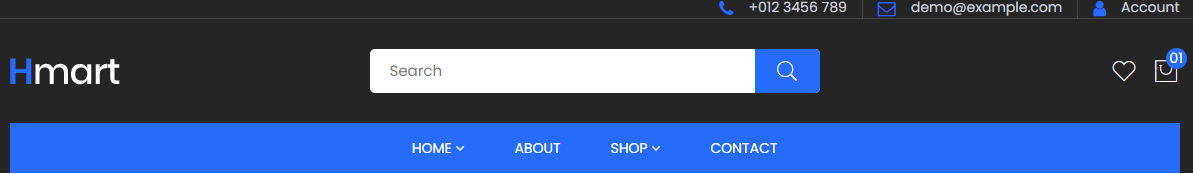


Рисунок 4.1 - Прототип navbar’a

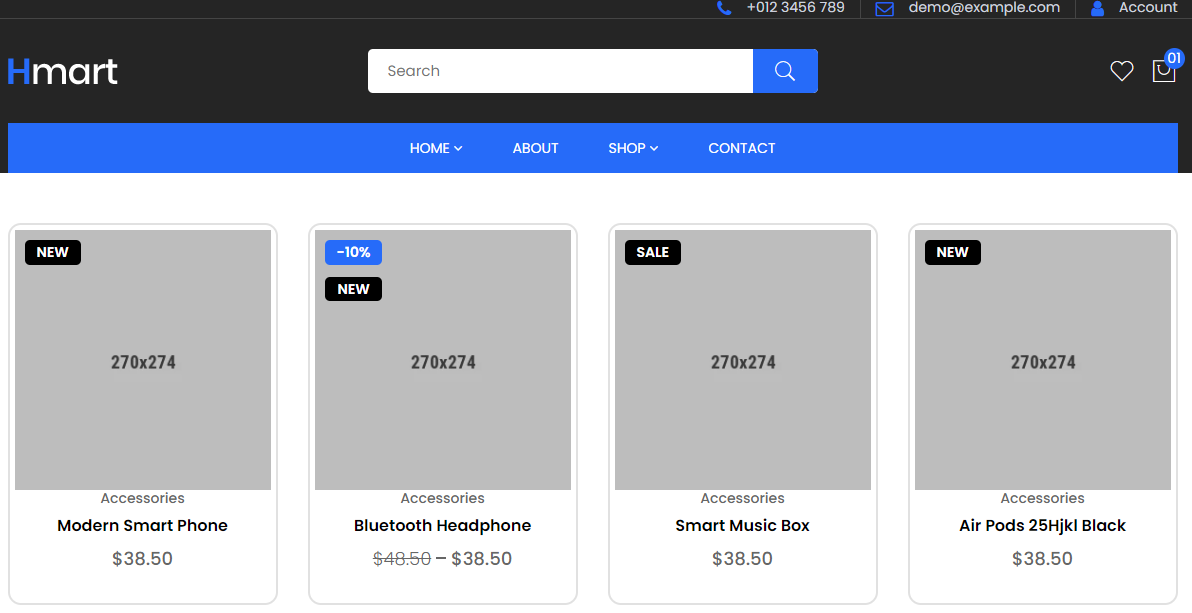


Рисунок 4.2 – Прототип главной страницы

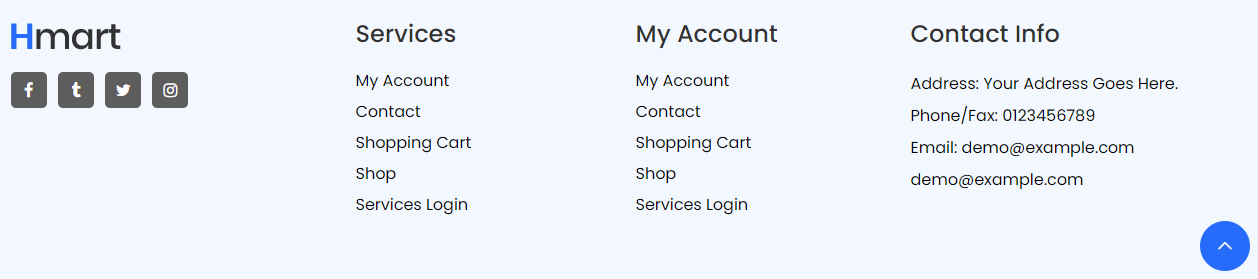


Рисунок 4.3 – Прототип footer’a

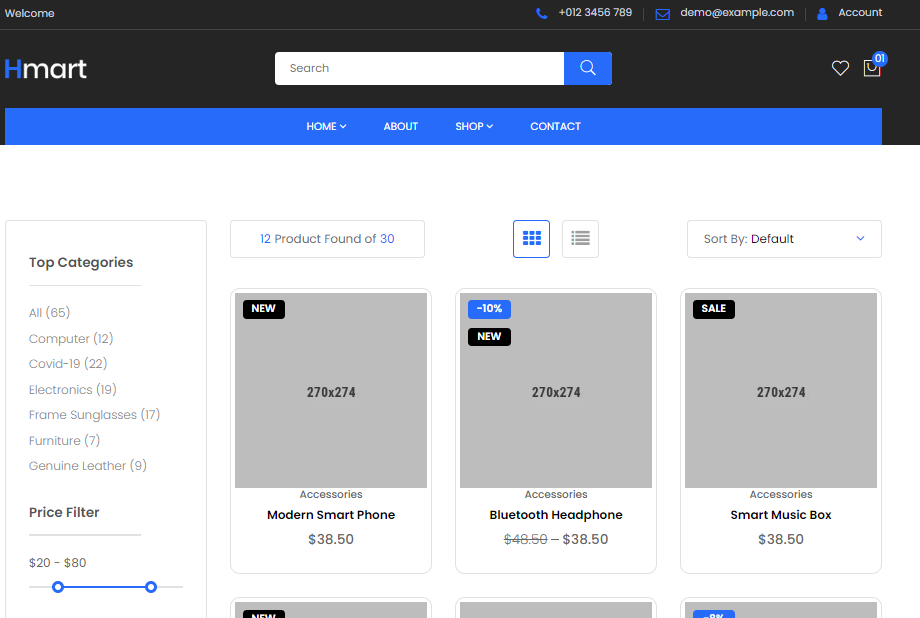


Рисунок 4.4 – Прототип страницы всех товаров (плитка)

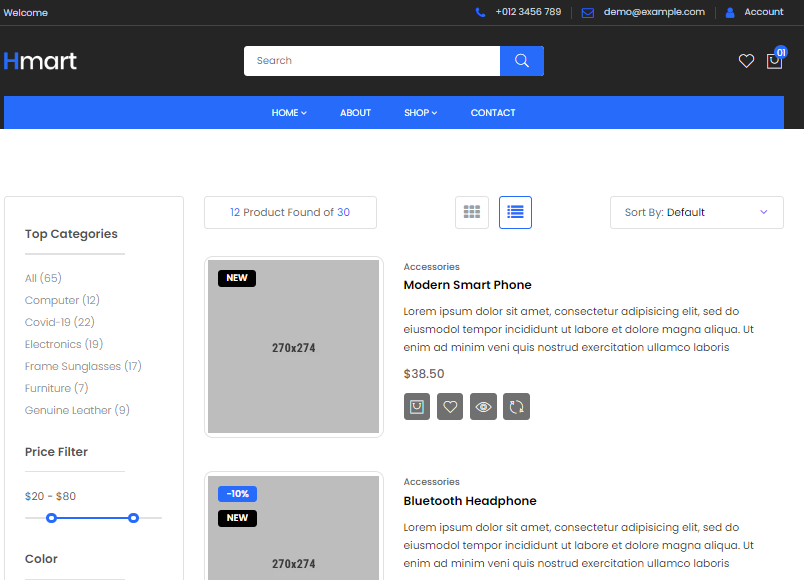


Рисунок 4.5 – Прототип страницы всех товаров (список)

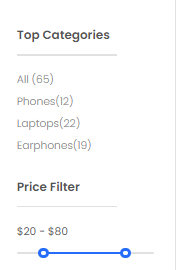
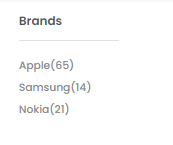
 

Рисунок 4.6 – Прототип фильтров на странице всех товаров

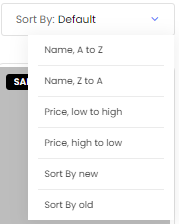


Рисунок 4.7 – Прототип списка сортировок товаров

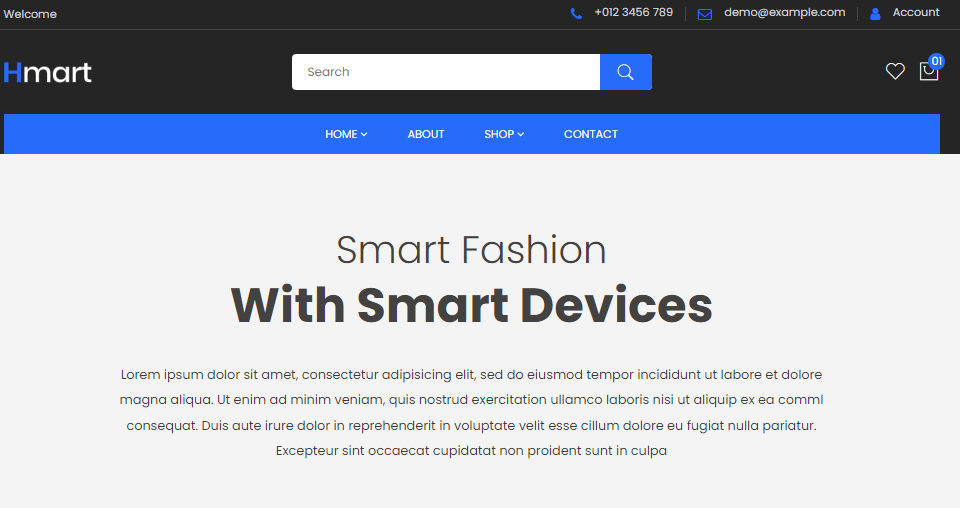


Рисунок 4.7.1 – Прототип страницы “О нас”

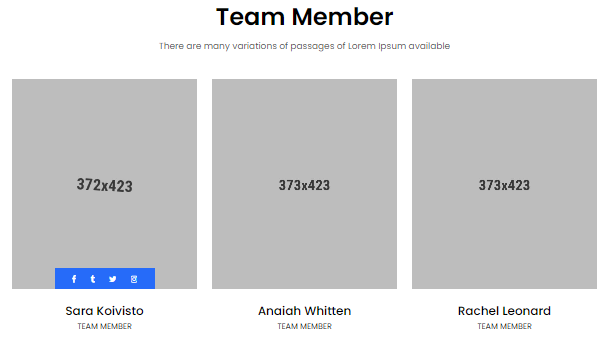


Рисунок 4.7.2 – Прототип страницы “О нас”

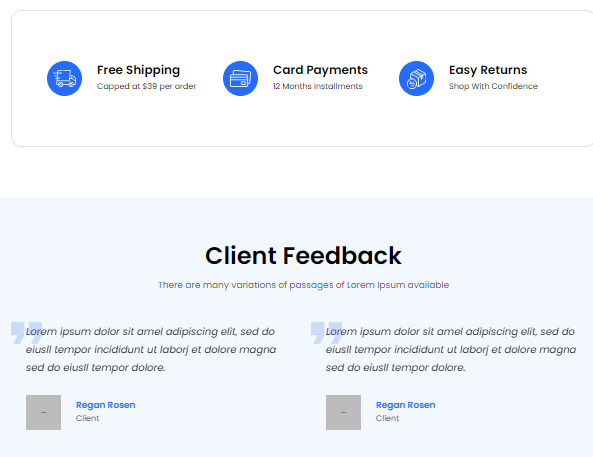


Рисунок 4.7.3 – Прототип страницы “О нас”

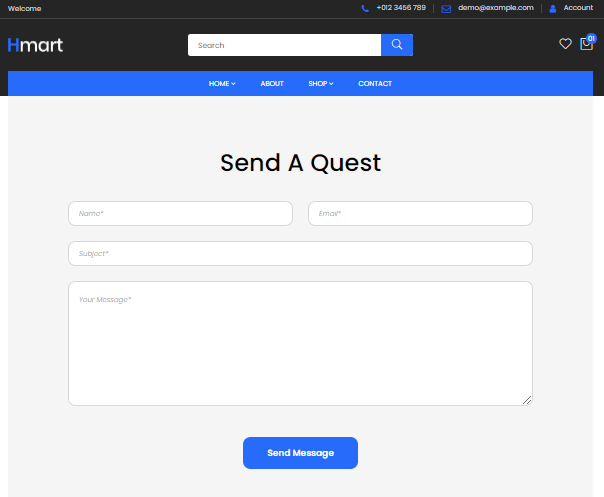


Рисунок 4.8.1 – Прототип страницы “Связь”. Форма обратной связи

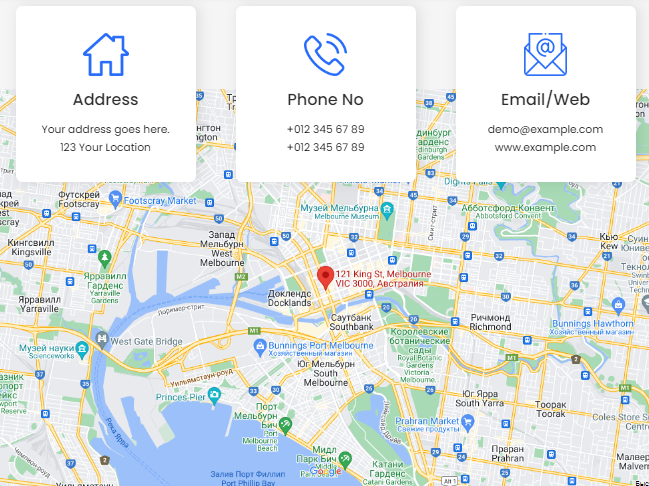


Рисунок 4.8.2 – Прототип страницы “Связь”

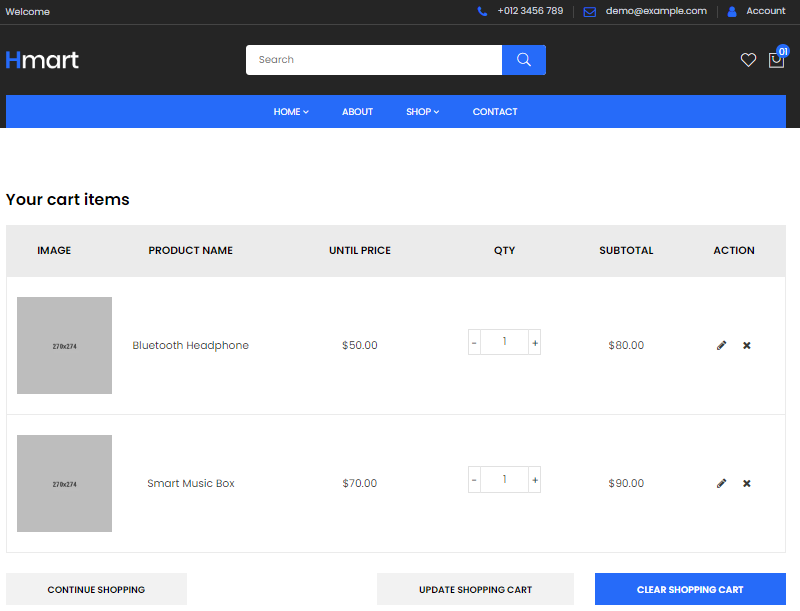


Рисунок 4.9 – Прототип страницы “Корзина”

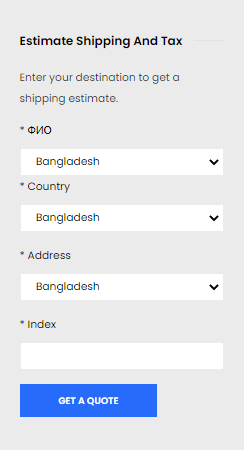


Рисунок 4.10 – Прототип формы заполнения адреса доставки

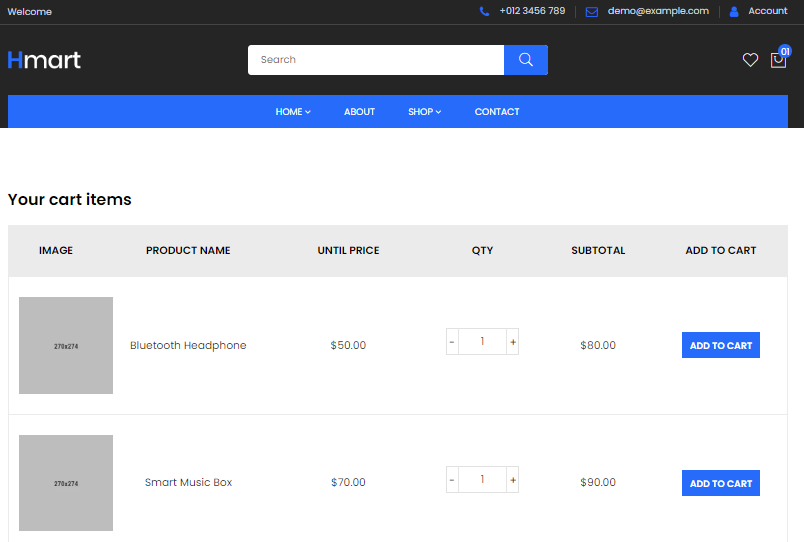


Рисунок 4.11 – Прототип страницы “Избранное”

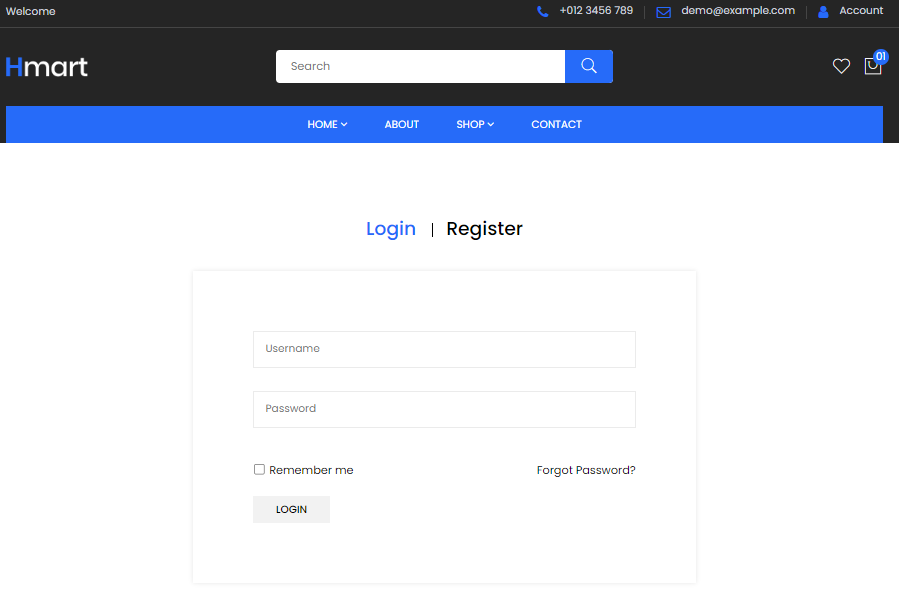


Рисунок 4.12 – Прототип страницы “Вход”

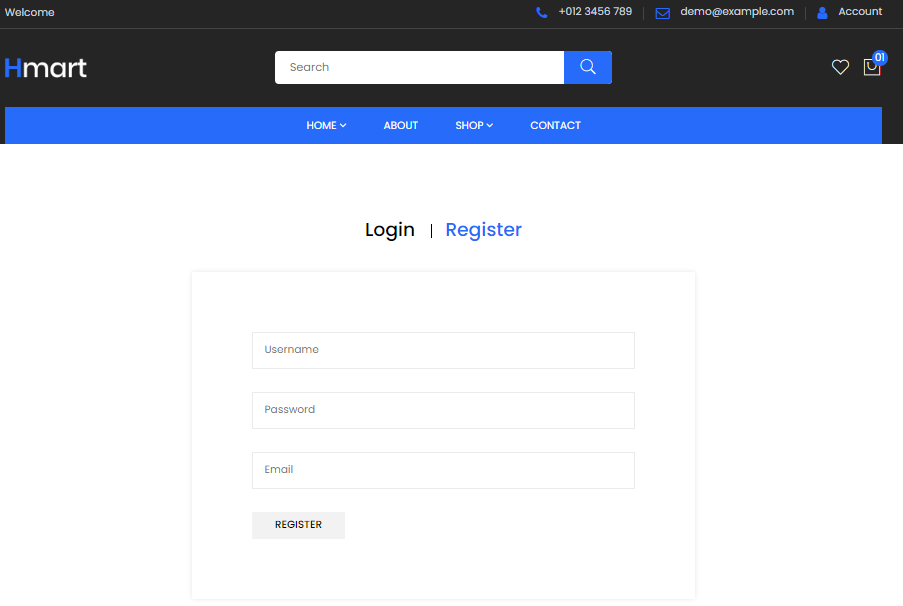


Рисунок 4.13 – Прототип страницы “Регистрация”

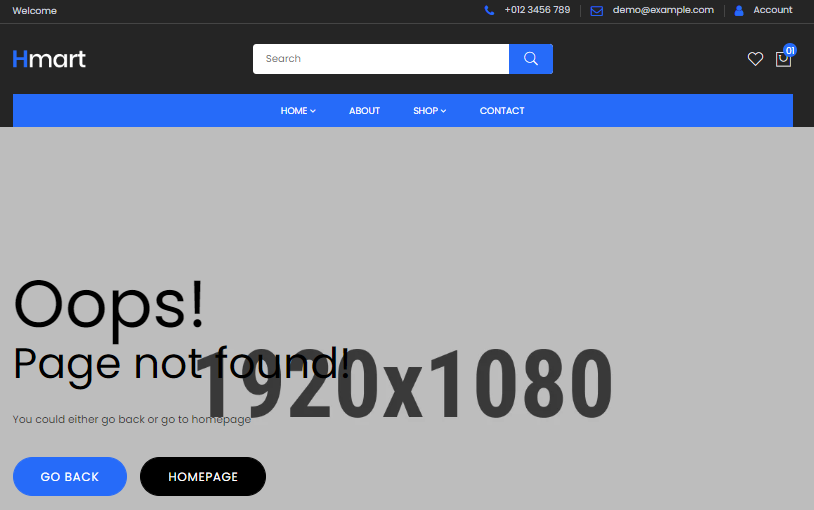


Рисунок 4.14 – Прототип страницы с ошибкой

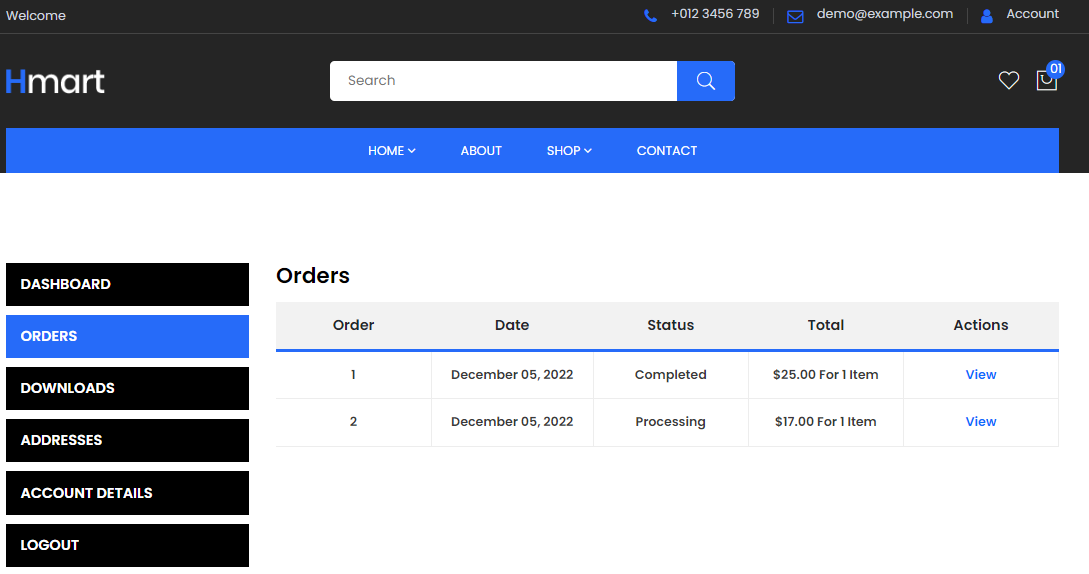


Рисунок 4.15.1 – Прототип страницы “Личный кабинет”. Список заказов

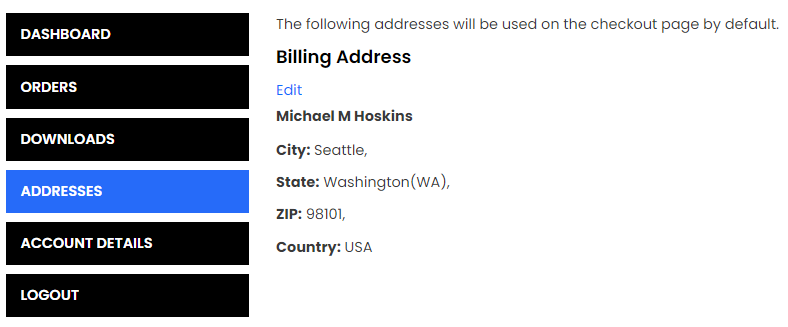


Рисунок 4.15.1 – Прототип страницы “Личный кабинет”. Сохранённые адреса

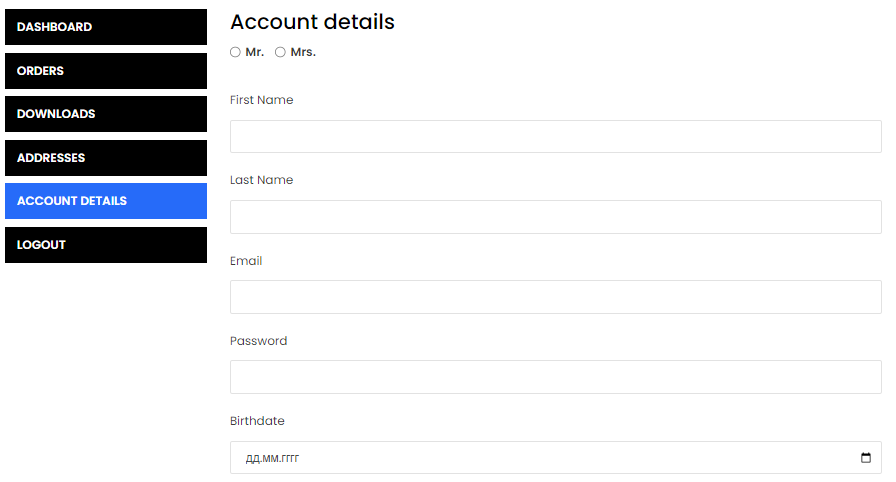


Рисунок 4.15.1 – Прототип страницы “Личный кабинет”. Информация об аккаунте

# ЗАКЛЮЧЕНИЕ

В результате проделанной работы был создан прототип веб-сайта «Интернет-магазин продажи электронных товаров».

Входе курсовой работы был проведен анализ предметной области, описана архитектура web-приложения, были составлены требования, на основании которых был разработан прототип будущего сайта. Созданный прототип веб-сайта полностью соответствует требованиям, предъявленным к проекту в требованиях.

При создании прототипа сайта были использованы самые популярные технологии, такие как HTML, CSS, JavaScript. Полученный сайт является адаптивным и имеет четкую логическую структуру, что без особого труда позволит привязать данный веб-сайт к какой-либо системе управления сайтами, использовать и развивать сайт в дальнейшем.

Разработанный прототип веб-сайта имеет понятный и простой пользовательский интерфейс. Реализованный сайт позволяет отображать информацию о товарах, о категориях товаров, позволяет клиентам добавлять товары в корзину, оформлять заказ.

Веб-сайт «Интернет-магазин продажи электронных товаров» поможет организации, желающей продавать свои товары через Интернет, расширить свою клиентскую базу и увеличить прибыль.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. w3school [Электронный ресурс]: w3c - Интернет-сайт - Режим доступа: http://www.w3schools.com/html/html5\_intro.asp

2. w3school [Электронный ресурс]: w3c - Интернет-сайт - Режим доступа: http://www.w3schools.com/css/css3\_intro.asp

3. The JavaScript Source [Электронный ресурс]: Developer JavaScript - Интернет-сайт - Режим доступа: http://www.javascriptsource.com

4. Bootstrap.ru [Электронный ресурс]: Bootstrap - Интернет-сайт - Режим доступа: http://bootstrap-ru.com/index.php

5. jQuery [Электронный ресурс]: The jQuery Foundation - Интернет-сайт - Режим доступа: http://jquery.com/

6. Django [Электронный ресурс]: Django - Интернет-сайт - Режим доступа: https://docs.djangoproject.com/en/4.1/