1. **Анализ задачи**
   1. **Постановка задачи**
      1. **Организационно-экономическая сущность задачи**

Наименование задачи: разработка интернет ресурса для продажи книг из частных коллекций «BookSpace».

Цель разработки: создание удобной онлайн-платформы для продажи книг с широким ассортиментом и удобным интерфейсом для пользователей.

Назначение: ПП предназначен для клиентов, желающих приобрести интересные книги, и администраторов магазина для работы с информацией о товарах и клиентской базой.

Периодичность использования: ежедневно, с возможностью посещения магазина в любое время суток.

Источники и способы получения данных: информация о магазине книг «BookSpace», его ассортименте, акциях, услугах, а также данные от поставщиков и издателей для пополнения каталога.

Информационная связь с другими задачами: проект тесно связан с задачами администрирования интернет-магазина, клиентской базой и системой обработки заказов.

Обзор существующих аналогичных ПП: анализ аналогичных платформ, таких как "Лабиринт", "Book24" и "Читай-город", показывает востребованность удобного интерфейса с функцией поиска и фильтрации, отзывов, а также наличия дополнительных сервисов, таких как персонализированные рекомендации, программа лояльности и система акций. Эти решения помогают повысить удобство и вовлеченность пользователей, что также может быть полезно для разработки функционала «BookSpace».

* + 1. **Функциональные требования**

Описание перечня функций и задач, которые должно выполнять будущее ПП:

Гость:

* просмотр каталога товаров и поиск с возможностью фильтрации: каждый товар будет иметь фото, названия, описание, цену, жанр, издательство. Поиск будет осуществляться в отдельной строке в шапке сайта или приложения. Фильтры будут по категориям: цена, жанр, издательство;
* регистрация и аутентификация гостя: пользователь при регистрации вводит свое ФИО, номер телефона, адрес, данные карты, придумывает пароль, электронную почту. При аутентификации вводит свою почту и пароль.

Пользователь – все те же функции, что может выполнять гость, но также появляется:

* авторизация;
* покупка в интернете;
* доступ к форме добавления книги на сайт;
* создание и редактирование профиля;
* выход из личного кабинета.

Администратор:

* управление контентом: добавление, удаление и изменение жанров и категорий;
* управление пользователями: просмотр, блокировка и отправка уведомлений пользователям.
  + 1. **Описание входной информации:**
* данные о товарах (фото, названия, описание, цену, жанр, издательство)**;**
* данные о клиентах (ФИО, номер телефона, адрес, данные карты, пароль, электронная почта)**;**
* данные заказа (количество и др.)**.**
  + 1. **Описание выходной информации:**
* подтверждение заказа, информация о статусе заказа, квитанция о покупке**;**
* формы представления: электронные письма с информацией о заказе;
* периодичность и сроки представления: моментально после оформления заказа.
  + 1. **Перечень условно-постоянной информации:**
* категории товаров**;**
* цены.
  + 1. **Нефункциональные (эксплуатационные) требования:**

Требования к применению:

Пользовательский интерфейс должен быть интуитивно понятным и привлекательным для пользователей всех возрастов.

Требования к производительности:

Время загрузки страницы не должно превышать 5 секунд.

Система должна поддерживать одновременную работу не менее 500 активных пользователей без значительного снижения производительности.

Время ответа сервера на запросы должно быть меньше 200 миллисекунд для обеспечения отзывчивости интерфейса.

Требования к реализации:

Для разработки следует использовать современные технологии для обеспечения эффективности и масштабируемости.

Все данные должны храниться в защищенной базе данных, с шифрованием конфиденциальной информации для обеспечения безопасности пользовательских данных.

Приложение должно быть развернуто на облачной инфраструктуре с высокой доступностью и автоматическим масштабированием для обеспечения непрерывной работы.

Требования к надежности:

Система должна быть способна обрабатывать сбои и восстанавливаться автоматически без потери данных.

Допустимое время простоя системы в случае сбоя не должно превышать 1 часа в месяц.

Регулярные резервные копии данных должны создаваться ежедневно и храниться в надежном и защищенном месте.

Требования к интерфейсу:

Система должна быть доступна через различные устройства, включая компьютеры, планшеты и мобильные телефоны.

Интерфейс должен быть совместим с различными браузерами, такими как Google Chrome, Mozilla Firefox и Microsoft Edge.

На главной странице будет возможность просмотра каталога товаров с фильтрацией, просмотр информации о местонахождении магазина и время его работы, просмотр соц. сетей магазина и ссылка на установку приложения.

* 1. **Диаграмма вариантов использования**

Диаграмма вариантов использования выглядит следующим образом (приложение А).

* 1. **Выбор стратегии разработки и модели жизненного цикла**

Для разработки веб-ресурса «Интернет магазина «BookSpace»» следует выбрать стратегию разработки и модель жизненного цикла. Осуществляем выбор посредством составления таблиц:

Таблица 1 - Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да |  |  |  |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да |  |  |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет |  |  |
| 4. | Нужно ли демонстрировать требования с целью их определения? |  |  | Да |  | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет |  | Нет |  |  |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? |  |  |  | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? |  |  | Да | Да | Да | Да |

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды разработчиков  проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет |  |  |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? |  |  | Нет | Нет | Нет |  |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет |  |  |  |
|  |  |  |  |  |  |  |  |
| Продолжение таблицы 2 | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? |  |  | Нет |  | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да |  |  |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да |  | Да | Да | Да |

Таблица 3 – Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да |  | Да |  | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет |  |  |  |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет |  | Нет |  | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? |  |  |  |  | Да | Да |

Таблица 4 – Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Разрабатыва ли в проекте продукт для организации направления? | Нет | Нет | Нет |  |  |  |
| Продолжение таблицы 4 | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2. | Будет ли проект являться расширением существующей системы? |  |  |  |  | Нет | Нет |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет |  |  |  |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да |  | Да |  | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? |  | Да |  | Да |  | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? |  |  |  | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет |  |  |  |
| 8. | Является ли график сжатым? | Нет | Нет |  |  |  |  |
| 9. | Предполагается ли повторное использование компонентов? |  |  | Да | Да | Да | Да |
| 10. | Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? |  |  |  |  | Да | Да |

В итоге заполнения таблиц 1-4 наиболее подходящим жизненным циклом является V-образная модель

* 1. **Инструменты разработки**

Для разработки данного проекта будут использованы языки программирования такие как: HTML, CSS, JavaScript и PHP, каждый из которых выполняет свою важную функцию в разработке веб-сайта.

HTML (HyperText Markup Language) будет использоваться для создания структуры веб-страниц. С помощью HTML можно определять такие элементы, как заголовки, абзацы, списки, изображения и ссылки. Этот язык позволяет упорядочить контент таким образом, чтобы он был логически структурирован и доступен для пользователей и поисковых систем.

CSS (Cascading Style Sheets) используется для оформления и визуального представления веб-страниц. CSS позволяет задать стили для различных элементов HTML, таких как цвет текста, шрифты, отступы, выравнивание, анимации и макеты. Благодаря CSS сайт будет выглядеть привлекательно и будет адаптирован под различные устройства, включая мобильные телефоны и планшеты.

JavaScript применяется для добавления интерактивности и динамичности на веб-сайт. С помощью JavaScript можно создавать скрипты, которые реагируют на действия пользователей, такие как клики по кнопкам, ввод данных в формы или перемещение мыши. Этот язык также позволяет работать с данными, осуществлять валидацию форм, загружать контент без перезагрузки страницы (AJAX) и многое другое.

PHP (Hypertext Preprocessor)используется для серверной части разработки. PHP позволяет создавать динамические веб-страницы, взаимодействовать с базами данных, обрабатывать формы, управлять сессиями и реализовывать другие серверные задачи. Благодаря PHP, сайт будет не только интерактивным, но и функциональным, с возможностью работы с данными и выполнения сложных логических операций на сервере.

Таким образом, использование этих четырех языков позволит создать функциональный, эстетически приятный и интерактивный веб-сайт, который будет удобен для пользователей и соответствовать современным стандартам веб-разработки.

* 1. **Разработка плана работы над проектом**

Для разработки данного проекта будет разработан план работы над проектом с помощью диаграммы Ганта (приложение Б).

1. **Проектирование задачи**
   1. **Разработка структуры сайта, системы меню, навигации**

Для разработки структуры сайта продажи книг будет создана диаграмма, отражающая взаимосвязь между ключевыми страницами: "Главная", "Каталог", "Корзина" и "Отзывы". Будет разработана удобная система меню и навигации, обеспечивающая быстрый доступ к категориям книг, фильтрам, поиску и личному кабинету, что значительно улучшит пользовательский опыт и облегчит процесс покупок (приложение В).

* 1. **Разработка UML-диаграмм**

В рамках разработки проекта будут созданы UML-диаграммы, такие как диаграмма деятельности, диаграмма объектов и диаграмма классов. Диаграмма деятельности отразит последовательность действий пользователей на сайте, диаграмма объектов покажет структуру хранения данных, связи между таблицами и их атрибуты, а диаграмма классов покажет структуру системы, взаимодействие между классами, их атрибуты и методы, а также связи, такие как ассоциации, агрегации и наследование. Эти диаграммы помогут визуализировать архитектуру проекта и упростят процесс разработки (приложение Г).

* 1. **Разработка пользовательского интерфейса**

Пользовательский интерфейс был разработан в приложении «Figma» (приложение Д).

Дизайн сайта выполнен в современном минималистичном стиле с использованием мягких цветовых переходов, ярких акцентов и четкой структуры. Основной фон имеет градиент, плавно переходящий от светло-розового (#FFC0CB) в верхней части к голубому (#ADD8E6) и фиолетовому (#D8BFD8) в нижней, создавая приятный и гармоничный визуальный эффект. Навигационная панель выполнена на светло-розовом фоне (#FADADD), а текст в меню представлен в чёрном (#000000) или тёмно-сером (#333333) цвете. При наведении на элементы меню используется подчёркивание или изменяется текста для интерактивности.

Заголовок страницы "Книги" выделяется крупным жирным шрифтом чёрного цвета (#000000) и размещён по центру, чётко контрастируя с фоновым градиентом. Карточки книг расположены в адаптивной сетке и имеют белый фон (#FFFFFF), лёгкую тень (#D3D3D3) для создания объёма и закруглённые края. Названия книг выделены жирным чёрным шрифтом, а текст описания выполнен в тёмно-сером цвете (#555555). Под каждой карточкой находится кнопка яркого цвета фуксии (#FF00FF) с белым текстом (#FFFFFF) и закруглёнными краями, что делает их заметными и удобными для взаимодействия.

Шрифт сайта современный, без засечек, например, Arial, Roboto или Open Sans. Размеры текста варьируются: заголовки крупные, от 24 до 32 пикселей, а описание и дополнительный текст имеют размер около 16 пикселей. Вся структура сайта организована с равномерными отступами между элементами, а сетка адаптируется к различным размерам экранов, переходя на две или одну колонку при уменьшении ширины окна.

Яркие акценты, такие как кнопки цвета фуксия, сочетаются с мягкими нейтральными фоновыми цветами, подчёркивая важные элементы интерфейса. Дизайн создаёт удобный пользовательский опыт, сохраняя визуальную простоту и эстетическую привлекательность.

1. **Реализация**
   1. **Создание сайта**

Процесс создания сайта включает в себя несколько этапов, начиная с определения целей и задач. Основной целью является создание удобной платформы для продажи книг из домашних коллекций. Среди основных задач выделяются обеспечение простого и интуитивно понятного интерфейса для покупателей и продавцов, возможность добавления и редактирования информации о книгах, а также реализация функционала корзины и оформления заказа.

Для разработки сайта выбраны следующие технологии: HTML5 и CSS3 для создания структуры и стилей страниц, JavaScript (включая такие библиотеки, как jQuery или Bootstrap) для добавления интерактивных элементов, PHP для обработки серверной логики и MySQL для хранения данных о книгах, пользователях и заказах.

Разработка сайта начинается с проектирования интерфейса, включая создание макета главной страницы, страниц каталога книг, корзины и личного кабинета пользователя. На следующем этапе создается структура сайта с использованием HTML-кода, который формирует основные страницы: главную страницу, каталог книг, страницу отдельной книги, корзину и личный кабинет пользователя.

После этого разрабатывается серверная часть, где на PHP пишутся скрипты для обработки запросов, управления базой данных и реализации авторизации пользователей. Далее выполняется интеграция базы данных: создаются таблицы в MySQL для хранения информации о книгах, пользователях и заказах.

Тестирование и отладка являются обязательной частью процесса. На этом этапе проводится функциональное и пользовательское тестирование, исправляются выявленные ошибки и оптимизируется производительность сайта.

Завершающий этап включает развертывание сайта. Настраивается хостинг на локальном сервере OSPanel, после чего сайт публикуется на удаленном сервере. После развертывания проводится финальная проверка работоспособности всех элементов.

* 1. **Руководство программиста**
     1. **Организация данных**

Определив атрибуты для каждой сущности и установив связи между ними, получили следующую схему данных (рисунок 1).

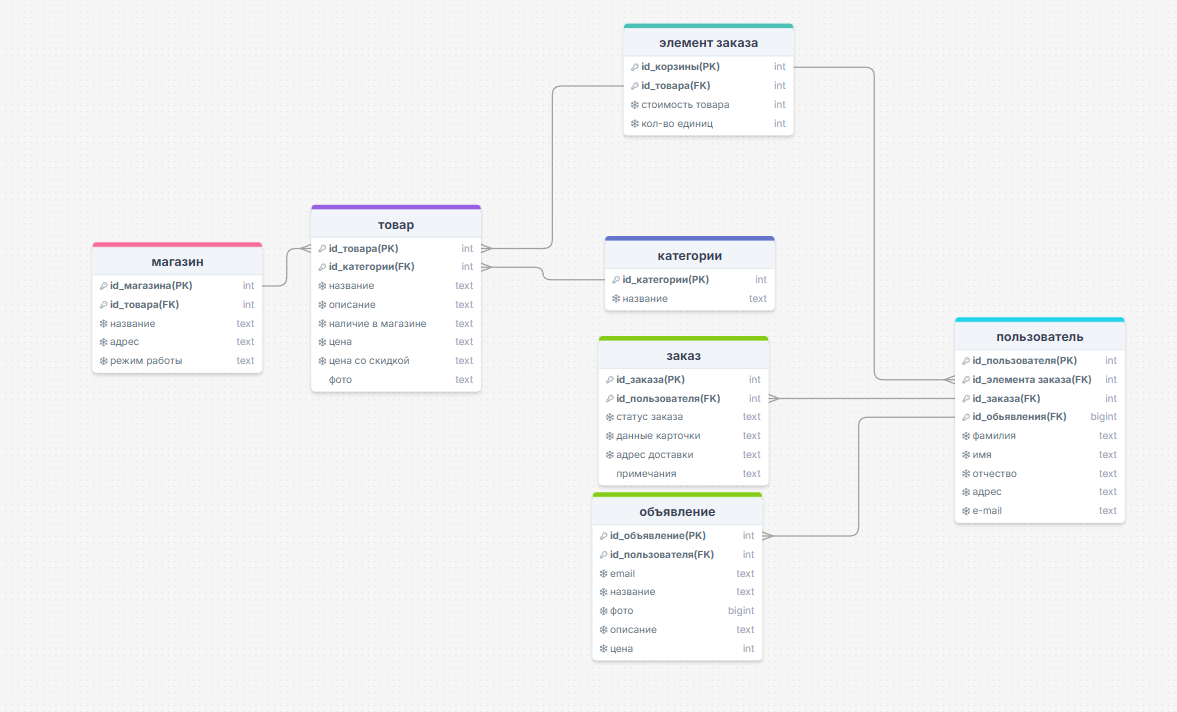


Рисунок 1 – Схема-данных

В данной БД представлено 7 сущностей. Описание каждой из них представлено в таблице 5.

Таблица 5 – Сущности

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Сущность | Имя поля | Тип данных | Описание |
| 1 | 2 | 3 | 4 | 5 |
| 1 | Магазин | ID\_магазина | Числовой | Содержит информацию о магазине |
| ID\_товара | Числовой |
| Название | Текстовый |
| Адрес | Текстовый |
| Режим работы | Текстовый |
| 2 | Товар | ID\_товара | Числовой | Содержит информацию о товарах |
| ID\_категории | Числовой |
| Название | Текстовый |
| Описание | Текстовый |
| Наличие в магазине | Текстовый |
| Цена | Текстовый |
| Цена со скидкой | Текстовый |
| Фото | Текстовый |
| 3 | Элемент заказа | ID\_корзины | Числовой | Содержит информацию о заказе |
| ID\_товара | Числовой |
| Количество единиц | Числовой |
| Стоимость товара | Числовой |
|  | | | | |
| Продолжение таблицы 5 | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 4 | Категории | ID\_категории | Числовой | Содержит информацию о категориях |
| Название | Текстовый |
| 5 | Объявление | ID\_объявления | Числовой | Содержит информацию о акциях |
| ID\_пользователя | Числовой |
| Email | Текстовый |
| Название | Текстовый |
| Фото | Текстовый |
| Описание | Текстовый |
| Цена | Числовой |
| 6 | Заказ | ID\_заказа | Числовой | Содержит информацию о заказе |
| ID\_пользователя | Числовой |
| Статус заказа | Текстовый |
| Данные карточки | Текстовый |
| Адрес доставки | Текстовый |
| Примечания | Текстовый |
| 7 | Пользователь | ID\_пользователя | Числовой | Содержит информацию о пользователе |
| ID\_элемента заказ | Числовой |
| ID\_заказа | Числовой |
| ID\_объявления | Числовой |
| Фамилия | Текстовый |
| Имя | Текстовый |
| Отчество | Текстовый |
| Адрес | Текстовый |
| Email | Текстовый |

* + 1. **Структура программы**

При входе на сайт у нас доступны возможности просмотра: каталога, времени работы магазина, скидок, регистрация. Так же можно будет фильтровать товары, а также искать их.

После авторизации будет доступна работа в личном кабинете, опубликование объявлений для обмена и заказ товаров.

У администратора будут доступны управление: каталогом, акциями, скидками, заказами.

* 1. **Диаграмма компонентов**

Диаграмма компонентов предоставляет наглядное представление архитектуры системы, демонстрируя ключевые модули и их взаимодействие. Она отражает основные компоненты сайта, такие как интерфейс пользователя, серверная часть и база данных, а также их взаимосвязь.

Данная диаграмма позволяет визуализировать структуру системы, выявить взаимозависимости и определить области, требующие дальнейшей доработки (приложение Е).

1. **Тестирование**

Каждый тест выполнялся пошагово: сначала проверялась работа отдельных элементов интерфейса, таких как кнопки, формы и ссылки. Затем тестировались основные сценарии использования, включая добавление товара в корзину, оформление заказа и авторизация пользователей.

Для проверки корректности работы различных функциональных модулей использовались тестовые учетные записи пользователей и предварительно созданные данные, такие как фейковые заказы и информация о товарах. Все тесты выполнялись вручную в реальных условиях работы сайта с использованием популярных браузеров, включая Google Chrome и Mozilla Firefox.

В процессе тестирования выявлялись ошибки, которые записывались в отдельный журнал для дальнейшего исправления. После устранения ошибок тесты повторялись, чтобы убедиться в их корректной работе и отсутствии регрессионных багов (таблица 6).

1. **Руководство пользователя**
   1. **Запуск программы**

Открываем сайт «BookSpace».

* 1. **Инструкция по работе с программой**

После открытия мы попадаем на главную страницу. На ней расположен каталог товаров, фильтрация и категории (рисунок 2).

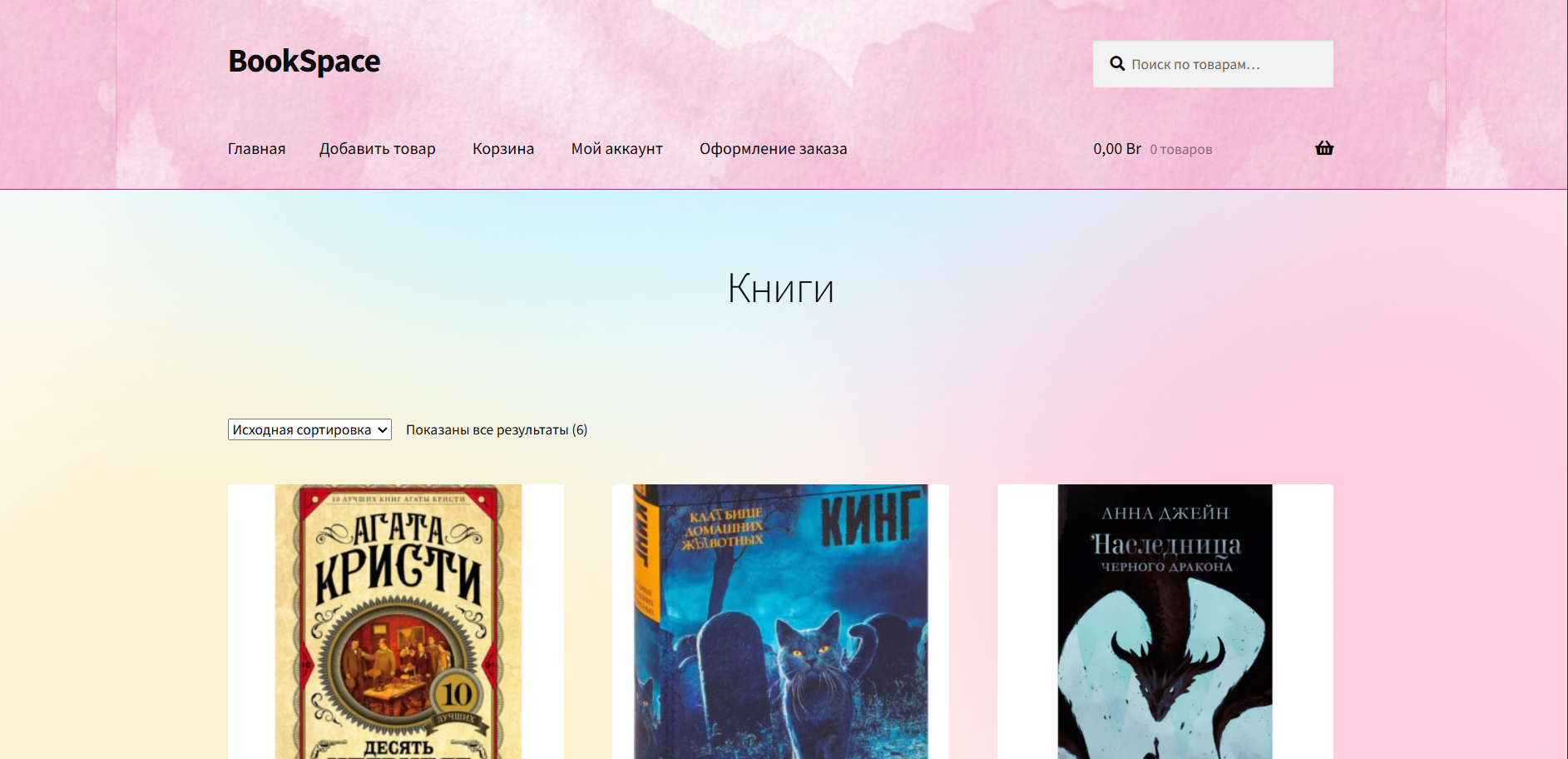


Рисунок 2 – Главная страница

При нажатии на товар можем просмотреть карточку товара (рисунок 3).

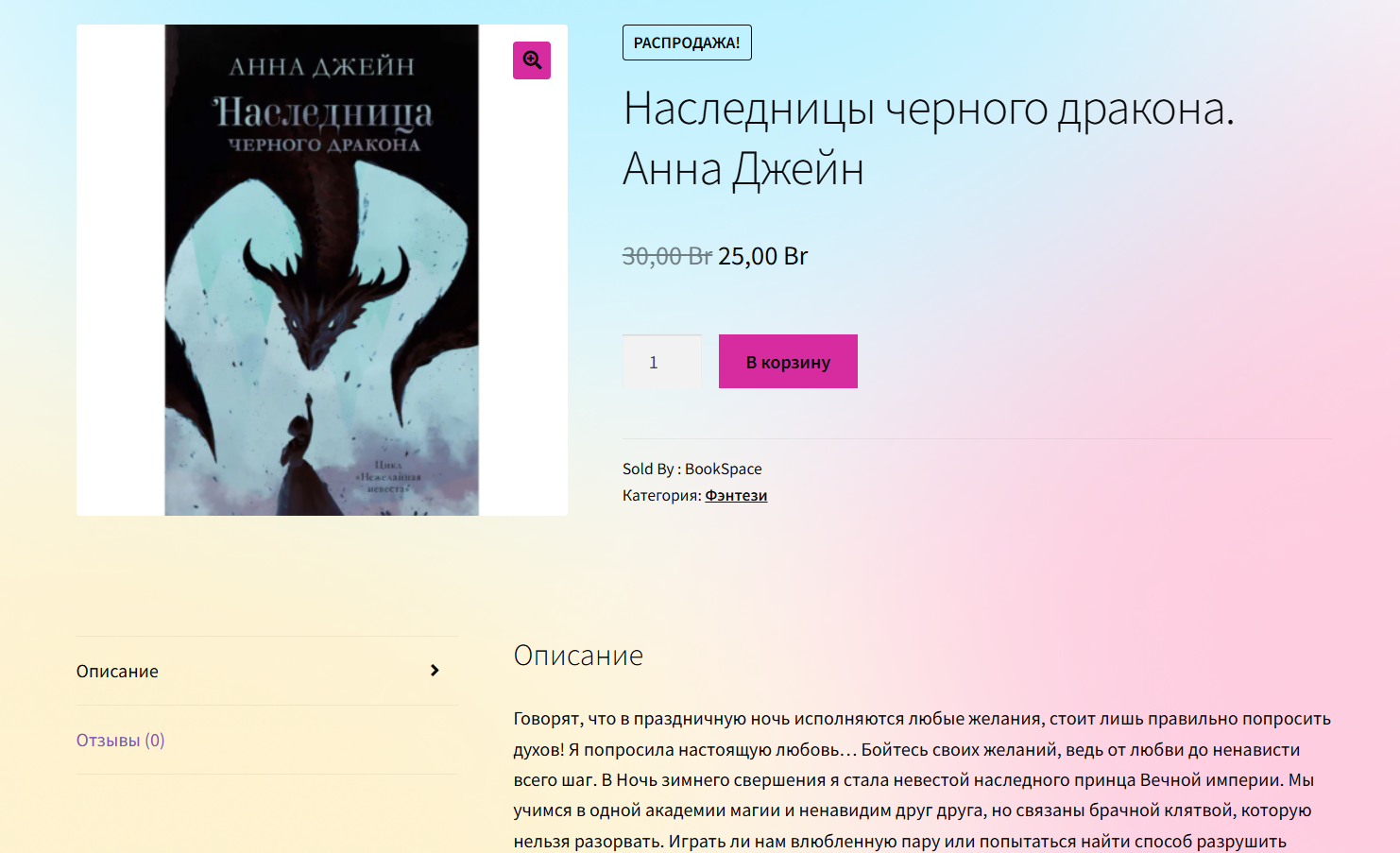


Рисунок 3 – Карточка товара

Можно нажать на иконку корзины в шапке сайта или просмотр корзины, чтобы просмотреть выбранный товар и заказать его (рисунок 4-5).

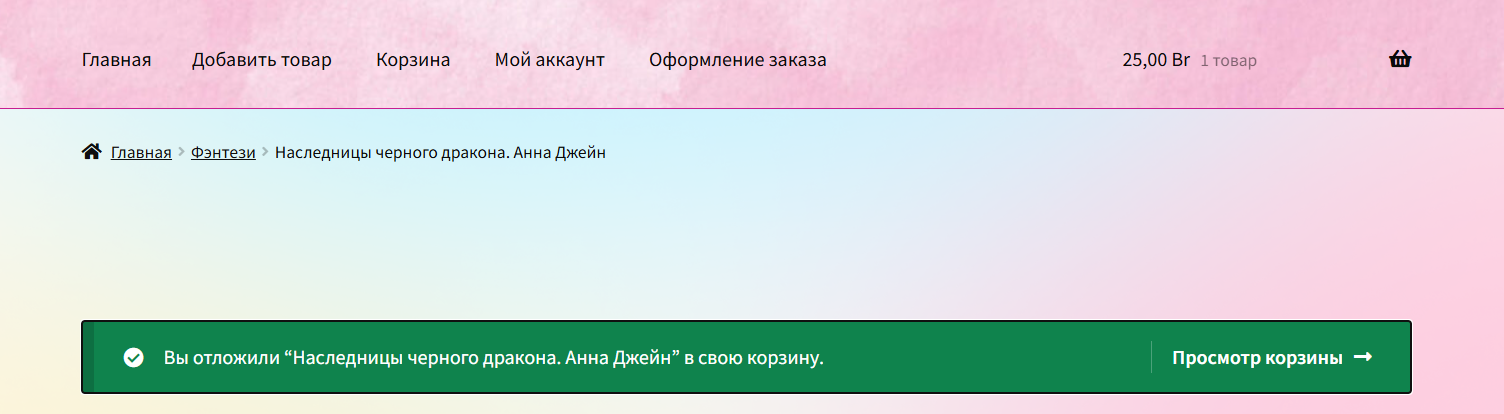


Рисунок 4 – Кнопки для просмотра корзины

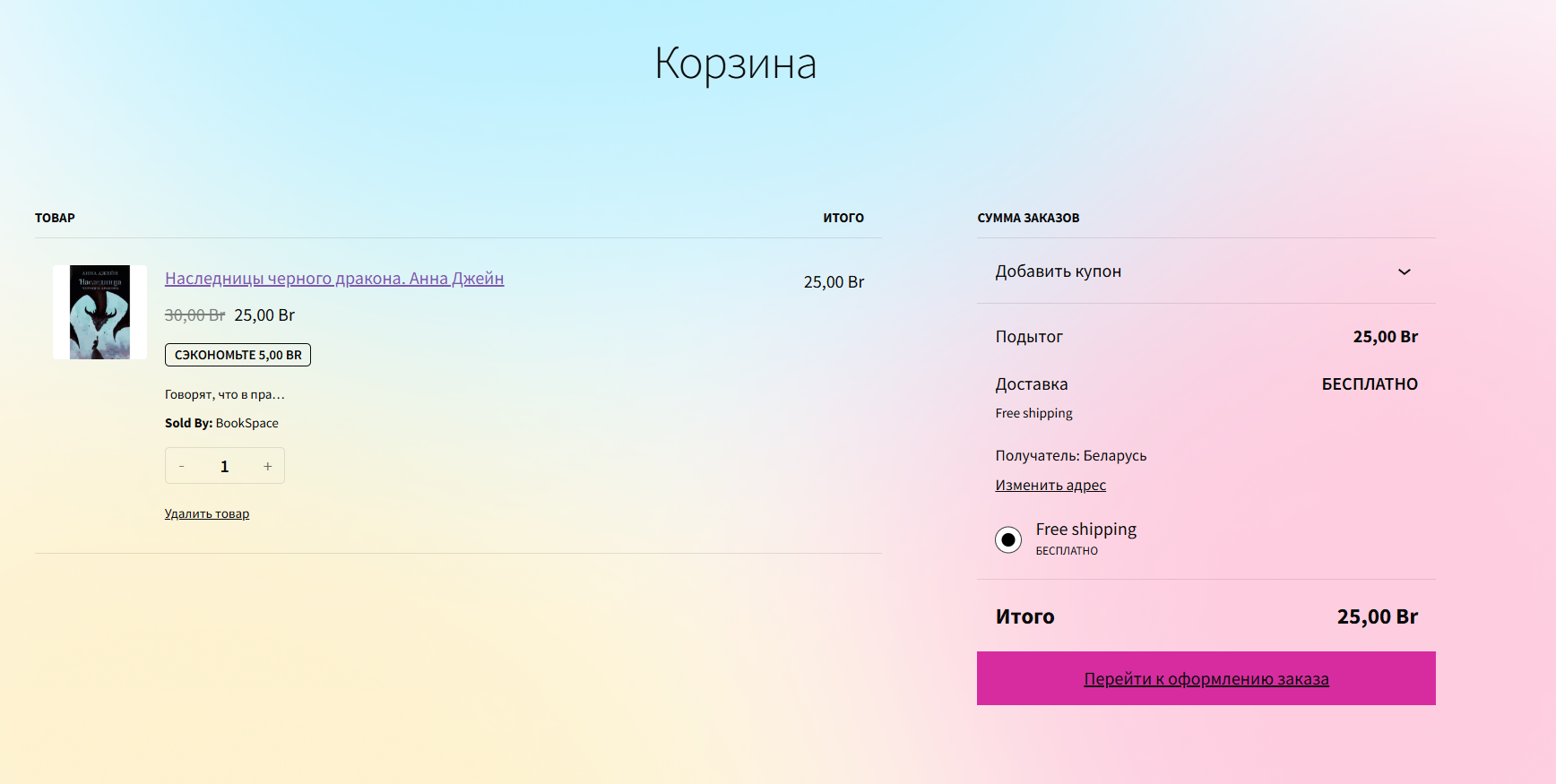


Рисунок 5 – Корзина

Так же с главной страницы можно перейти на вкладку «Мой аккаунт» и зарегистрироваться (рисунок 6).

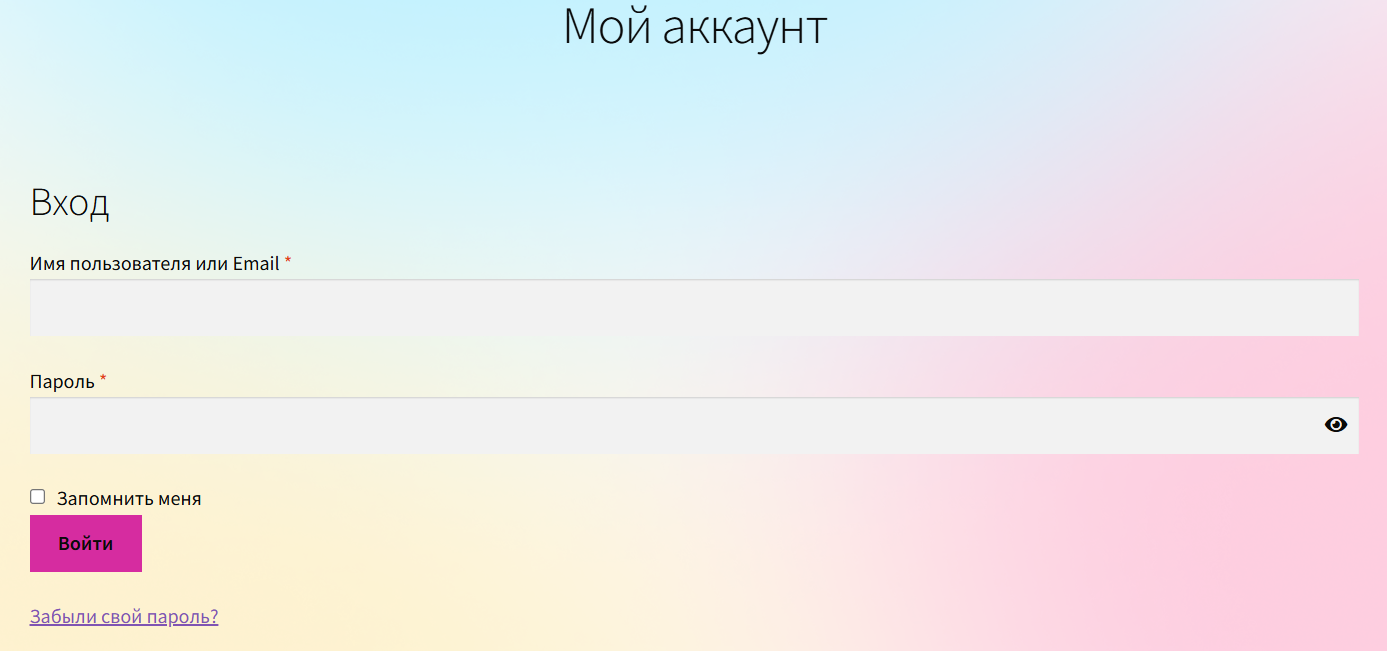


Рисунок 6 – Вкладка «Мой аккаунт»

Далее можно перейти на страницу «Добавить товар», после заполнения полей можно опубликовать свой товар на сайт (доступно после авторизации) (рисунок 7).

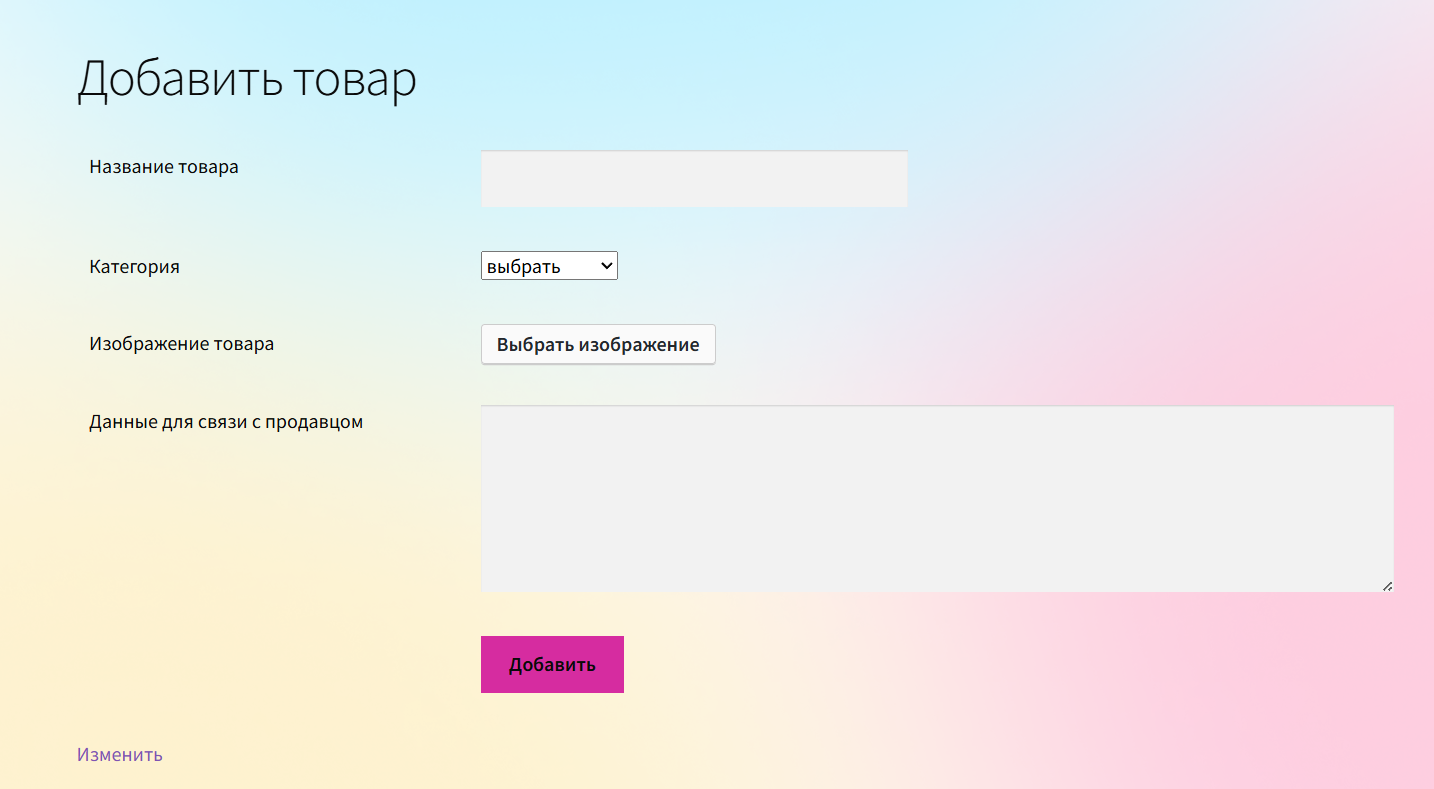


Рисунок 7 – Вкладка «Добавить товар»

* 1. **Завершение работы с программой**

Закрываем сайт «BookSpace».

**Заключение**

Разработка приложения на тему: «Разработка интернет ресурса для продажи книг из частных коллекций», направленная на создание удобного и функционального веб-ресурса для продажи книг, была выполнена с использованием системы управления контентом WordPress.

Для разработки сайта использовались следующие технологии и инструменты:

* HTML, CSS, JavaScript – для создания структуры, оформления и функциональности страниц;
* WordPress – для реализации структуры сайта и управления контентом;
* MySQL – для работы с базой данных;
* PhpMyAdmin – для управления базой данных;
* OSPanel – для локального тестирования и отладки сайта;
* Microsoft Word 2021 – для написания документации по проекту.

Использование данных технологий и инструментов позволило создать полноценный веб-сайт, включающий основные функции, такие как:

* регистрация и авторизация пользователей;
* добавление и редактирование объявлений о продаже книг;
* просмотр и фильтрация каталога книг;
* управление заказами;
* создание адаптивного интерфейса, удобного для работы с различных устройств.

Степень соответствия проектных решений заданию: сайт был выполнен в соответствии с поставленными требованиями.

Проект демонстрирует возможности современных веб-технологий для создания качественного и функционального продукта, который может быть дополнен и расширен в будущем для улучшения пользовательского опыта.

**Список использованных источников**

1. [https://wordpresslab.ru/themes/merchant/?utm\_source=yandex&utm\_medium= cpc&utm\_campaign=101914862](https://wordpresslab.ru/themes/merchant/?utm_source=yandex&utm_medium=%20cpc&utm_campaign=101914862);
2. <https://tobiz.net/?utm_source=yandexdirect&banner=14756043744&campaign_id=9208633>;
3. <https://wordpress.org>