

Приложение Д.....	35
Приложение Е.....	37
Приложение Ж.....	39
Приложение И	44
Приложение К.....	48
Приложение Л.....	52

Введение

На учебную практику по разработке и сопровождению программного обеспечения была поставлена задача разработать сайт на тему: «Сайт дизайнера мужской одежды «Pants for men»». Цель проекта заключается в создании удобного и функционального веб-сайта для продажи различных видов штанов с возможностью добавления принта онлайн. При разработке сайта важно было учесть потребности покупателей: обеспечить удобную навигацию, предоставить подробную информацию о товарах и организовать простой и безопасный процесс покупки. Сайт должен включать в себя каталог товаров, корзину покупок, добавление изображения и возможность онлайн-заказа.

Пояснительная записка состоит из нескольких разделов, которые подробно описывают разработку сайта.

Первый раздел, «Анализ задачи», определяет цели и задачи проекта, исследует предметную область, анализирует существующие решения и описывает входные и выходные данные. Раздел глубоко погружается в анализ существующих аналогов, выявляя их сильные и слабые стороны. В нем также исследуется периодичность использования аналогичных программных продуктов и рассматривается, как разрабатываемое приложение будет упрощать решение поставленной задачи по сравнению с существующими решениями.

В подразделе «Инструменты разработки» рассматриваются среда разработки, язык программирования и системные требования к аппаратным ресурсам. Раздел предоставляет детальное обоснование выбора конкретной среды разработки и языка программирования, а также описывает оптимальные и минимальные конфигурации технических средств, необходимых для корректного и эффективного функционирования приложения.

Раздел «Проектирование задачи» посвящен основным аспектам разработки программного продукта. Здесь описывается организация данных, дизайн пользовательского интерфейса, алгоритмы обработки информации и разработка справочной системы. В этом разделе представлен детальный анализ архитектуры приложения, структуры базы данных, а также описаны концепции пользовательского интерфейса, включая дизайн и эргономику программы.

Третий раздел, «Реализация», детально описывает элементы и объекты, используемые при реализации приложения, а также структуру и функции пользователя. В этом разделе описывается подробная реализация функций приложения, включая алгоритмы обработки данных, механизмы взаимодействия с пользователем и описание структуры кода.

В разделе «Тестирование» представлено полное и функциональное

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

тестирование программы, включая проверку всех пунктов меню и сценариев работы пользователя. Раздел подробно описывает методы тестирования, используемые в проекте, и предоставляет результаты проверки всех функциональных возможностей приложения.

В разделе «Руководство пользователя» будет описано назначение, область применения, среда функционирования данного программного продукта.

«Заключение» кратко формулирует задачу проекта, описывает полученные результаты, используемые методы и средства, а также степень автоматизации различных этапов разработки. В «Списке использованных источников» приведен список литературы, использованной при разработке проекта.

В разделе «Список использованных источников» будет приведён список используемых при разработке источников.

В приложении А будет представлена диаграмма вариантов использования.

В приложении Б будет представлена диаграмма Ганта.

В приложении В будет представлена структура сайта.

В приложении Г будет представлена диаграмма последовательности.

В приложении Д будет представлена диаграмма классов.

В приложении Е будет представлена модель данных.

В приложении Ж будут представлены тест-кейсы.

В приложении И будут представлены UX-прототипы.

В приложении К будут представлены UI-прототипы.

В приложении Л будет представлен листинг продукта.

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

1 Анализ задачи

1.1 Постановка задачи

Наименование задачи: сайт по продаже дизайнерской мужской одежды.

Цель разработки: создание сайта для заказа штанов онлайн.

Назначение: данные программные продукты разрабатываются для людей любого возраста, желающих заказать штаны в Интернете и нанести на них собственный принт.

Периодичность использования: по мере необходимости.

Источники и способы получения данных: информация пользователей для оформления заказа, информация о товаре.

Обзор существующих аналогичных ПП: для анализа рынка и выявления основных тенденций были рассмотрены несколько интернет-магазинов, специализирующихся на продаже одежды, в том числе и штанов. В качестве примеров можно привести Wildberries, Lamoda, а также сайты отдельных брендов. На этих сайтах присутствуют такие функции, как каталог товаров с возможностью фильтрации и сортировки, корзина покупок, поиск товаров. Однако анализ показал, что некоторые сайты имеют сложную навигацию и перегруженный интерфейс. В разрабатываемом интернет-магазине штанов будут реализованы аналогичные функции, но с учетом выявленных недостатков. Основной упор будет сделан на простоту и удобство использования, чтобы процесс покупки штанов был максимально быстрым и эффективным.

Функциональные требования к разрабатываемому интернет-магазину штанов включают в себя ряд возможностей для пользователей. Гости сайта смогут просматривать информацию о товарах, такую как модель, фотографии и доступные размеры. Также им будут доступны функции добавления штанов в корзину, удаление из корзины, изменение количества товаров и оформление заказа. Кроме того, гостям сайта будет предоставлена возможность персонализации штанов путем загрузки собственного принта в виде картинки, который будет нанесен на выбранную модель. Система должна предусматривать информацию о требованиях к формату загружаемого изображения. Администратор сайта сможет добавлять новые товары, редактировать информацию о существующих товарах (включая изменение цены, описания, фотографий и доступных размеров), удалять товары, просматривать заказы пользователей, управлять статусами заказов (например, «принят», «в обработке», «отправлен», «доставлен»).

Для корректной работы системы необходима определенная входная

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

информация. Это включает в себя название модели штанов и подробную информацию о них, например, описание, состав материала, особенности кроя.

В результате работы системы пользователь получает информацию о своем заказе, которая может включать номер заказа, состав заказа, выбранный способ доставки и оплаты, а также контактные данные.

В системе также будет храниться условно-постоянная информация. К ней относятся названия моделей штанов и их фотографии. Эта информация будет регулярно обновляться по мере поступления новых коллекций.

Эксплуатационные требования определяют условия использования и технические характеристики системы. Интернет-магазин должен помогать пользователям быстро находить и выбирать подходящие штаны. Для реализации фронтенда будут использоваться HTML, CSS и JavaScript. Серверная часть будет разработана на Java. Система должна быть надежной, с допустимым временем простоя не более 24 часов в год. Также необходимо предусмотреть механизм резервного копирования и восстановления данных. Дизайн интерфейса должен быть выполнен в коричнево-бежевых, белых, серых и черных тонах. Все основные разделы сайта должны быть доступны с главной страницы. Интерфейс должен быть интуитивно понятным и удобным для всех пользователей. На главной странице будут представлены фотографии и названия моделей штанов. При клике на изображение откроется страница с подробной информацией о товаре и возможностью добавления в корзину или оформления заказа. Сайт должен быть адаптивным и корректно отображаться на различных устройствах: компьютерах, планшетах и смартфонах.

1.2 Диаграмма вариантов использования

Диаграмма вариантов использования описывает функциональные возможности интернет-магазина штанов и взаимодействие с ним двух типов пользователей: Гость и Администратор. Гость, не зарегистрированный в системе пользователь, имеет доступ к основным функциям магазина. Он может просматривать подробную информацию о товарах, представленных в каталоге, включая описание, фотографии и цену. Выбрав понравившийся товар, Гость может добавить его в корзину. При добавлении в корзину Гость указывает необходимый размер и количество единиц товара. Если Гость передумал покупать какой-либо товар, он может удалить его из корзины. После того как все необходимые товары добавлены в корзину, Гость может оформить заказ, указав информацию для доставки и оплаты. Также Гость имеет возможность

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		7

просматривать информацию о товарах, уже находящихся в его корзине. Кроме того, магазин предоставляет возможность заказать штаны с индивидуальным принтом. Для этого Гость может загрузить собственное изображение, предварительно ознакомившись с требованиями к изображению принта.

Администратор, в отличие от Гостя, имеет полный доступ к управлению магазином. Он может добавлять новые товары в каталог, указывая всю необходимую информацию о них, такую как название, описание, цена, размеры и фотографии. Администратор также может удалять товары из каталога, если они больше не доступны для продажи. Редактирование информации о товаре также входит в обязанности Администратора, позволяя ему обновлять цены, описания и другие характеристики товаров. Администратор работает с заказами пользователей, просматривая их детали и управляя их статусами. Он может изменять статус заказа, например, отмечая его как «принят», «в обработке», «отправлен» или «выполнен». Для организации каталога товаров Администратор может создавать новые категории и удалять существующие. Работа с товарами и работа с заказами представляют собой комплексные функции, включающие в себя несколько подзадач, описанных выше. Таким образом, диаграмма вариантов использования наглядно демонстрирует распределение функциональности между Гостем и Администратором, предоставляя полное представление о возможностях системы для каждого типа пользователя.

Разработанная диаграмма вариантов использования представлена в приложении А.

1.3 Выбор стратегии разработки и модели жизненного цикла

Для разработки веб-ресурса заказа штанов онлайн важно правильно выбрать стратегию разработки и модель жизненного цикла. Этот выбор напрямую влияет на успех проекта, его сроки, бюджет и качество конечного продукта. Выбор осуществляется путем сравнительного анализа различных вариантов, но прежде необходимо понять значимость этого этапа. Правильная стратегия и модель жизненного цикла обеспечивают предсказуемость проекта, позволяя точно оценить сроки выполнения и необходимые финансовые ресурсы. Они также обеспечивают контроль над процессом разработки, позволяя отслеживать прогресс и вносить корректировки по мере необходимости. Гибкость в процессе разработки крайне важна, и выбранная модель должна предоставлять возможность адаптироваться к изменяющимся требованиям. Качество конечного продукта напрямую зависит от выбранных методологий;

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		8

систематическое тестирование является неотъемлемой частью процесса разработки надежного веб-ресурса. Эффективность использования ресурсов (времени, бюджета, команды разработчиков) достигается за счет четкой организации процесса разработки. Удовлетворенность заказчика является приоритетной задачей, и постоянное взаимодействие, а также получение обратной связи играют важную роль в достижении этой цели. Структурированный подход к разработке, определяемый выбранной стратегией и моделью, позволяет снизить риски проекта путем раннего выявления и решения потенциальных проблем. Таким образом, правильный выбор стратегии разработки и модели жизненного цикла является основой для успешного создания веб-ресурса.

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 определяемыми и реализуемыми?	<u>Да</u>	<u>Да</u>	<u>Да</u>	Нет	Нет	Нет
2.	Могут ли требования быть сформулированы в начале ЖЦ?	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>	Нет	Нет
3.	Часто ли будут изменяться требования на протяжении ЖЦ?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да
4.	Нужно ли демонстрировать требования с целью их определения?	Нет	Нет	<u>Да</u>	Нет	<u>Да</u>	<u>Да</u>
5.	Требуется ли проверка концепции программного средства или системы?	Нет	Нет	<u>Да</u>	Нет	<u>Да</u>	<u>Да</u>
6.	Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
7.	Нужно ли реализовать основные требования на ранних этапах разработки?	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>

Вычисления: 4 за каскадную, 4 за V-образную, 7 за RAD, 3 за инкрементную, 3 за быстрого прототипирования и 3 за эволюционную.

Итог: На основе результатов заполнения табл. 1 подходящей является RAD модель.

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.	Являются ли проблемы предметной области проекта новыми для большинства разработчиков?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да
2.	Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков?	Да	Да	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да
3.	Изменяются ли роли участников проекта на протяжении ЖЦ?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
4.	Является ли структура процесса разработки более значимой для разработчиков, чем гибкость?	Да	Да	<u>Нет</u>	Да	<u>Нет</u>	<u>Нет</u>
5.	Важна ли легкость распределения человеческих ресурсов проекта?	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>	Нет	Нет
6.	Приемлет ли команда разработчиков оценки, проверки, стадии разработки?	<u>Да</u>	<u>Да</u>	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>

Вычисления: 4 за каскадную, 4 за V-образную, 5 за RAD, 4 за инкрементную, 3 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 2 подходящей является RAD модель.

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.	Будет ли присутствие пользователей ограничено в ЖЦ разработки?	Да	Да	<u>Нет</u>	Да	<u>Нет</u>	Да
2.	Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
3.	Будут ли пользователи вовлечены во все фазы ЖЦ разработки?	Нет	Нет	<u>Да</u>	Нет	<u>Да</u>	Нет
4.	Будет ли заказчик отслеживать ход выполнения проекта?	Нет	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>

Вычисления: 1 за каскадную, 1 за V-образную, 3 за RAD, 0 за инкрементную, 3 за быстрого прототипирования и 1 за эволюционную.

Итог: На основе результатов заполнения табл. 3 подходящей является модель быстрого проектирования и RAD модель.

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.	Разрабатывается ли в проекте продукт нового для организации направления?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
2.	Будет ли проект являться расширением существующей системы?	Да	Да	Да	Да	<u>Нет</u>	<u>Нет</u>
3.	Будет ли проект крупно- или среднемасштабным?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
4.	Ожидается ли длительная эксплуатация продукта?	<u>Да</u>	<u>Да</u>	<u>Нет</u>	<u>Да</u>	<u>Нет</u>	<u>Да</u>
5.	Необходим ли высокий уровень надежности продукта проекта?	<u>Нет</u>	<u>Да</u>	<u>Нет</u>	<u>Да</u>	<u>Нет</u>	<u>Да</u>
6.	Предполагается ли эволюция продукта проекта в течение ЖЦ?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
7.	Велика ли вероятность изменения системы (продукта) на этапе сопровождения?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
8.	Является ли график сжатым?	<u>Нет</u>	<u>Нет</u>	Да	Да	Да	Да
9.	Предполагается ли повторное использование компонентов?	<u>Нет</u>	<u>Нет</u>	Да	Да	Да	Да
10.	Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да

Вычисления: 2 за каскадную, 3 за V-образную, 3 за RAD, 7 за инкрементную, 7 за быстрого прототипирования и 9 за эволюционную.

Итог: На основе результатов заполнения табл. 4 подходящей является эволюционная модель.

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является RAD модель.

1.4 Инструменты разработки

Для разработки данного проекта будут выбраны среды разработки VS

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		11

Code и IntelliJ IDEA, которые являются наиболее актуальными средами для создания приложений данного типа. VS Code, благодаря своей легкости, гибкости и огромному количеству расширений, отлично подойдет для разработки клиентской части и написания JavaScript-кода. IntelliJ IDEA, в свою очередь, зарекомендовала себя как мощная среда для разработки на Java, предоставляющая продвинутые инструменты для анализа кода, отладки и рефакторинга, что критически важно для разработки серверной части.

Для реализации интернет-магазина будет использован следующий набор технологий и инструментов:

Фронтенд (клиентская часть):

1 HTML (HyperText Markup Language) – для создания разметки сайта;

2 CSS (Cascading Style Sheets) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки;

3 JavaScript – задание реакции элементов на действия пользователя; прототипно-ориентированный сценарный язык программирования. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Бэкенд (серверная часть):

1 Java: язык программирования для реализации серверной логики, обработки данных и взаимодействия с базой данных. Выбор Java обусловлен его производительностью, надежностью и широким набором доступных библиотек и фреймворков. Будет использован фреймворк Spring, предоставляющий инфраструктуру для упрощения разработки и поддержки приложений.

2 База данных: PostgreSQL. Будет использоваться для хранения информации о товарах, пользователях, заказах и другой необходимой информации.

Иные инструменты, используемые при разработке и написании сопутствующей документации:

1 Postman – будет использоваться для отправки запросов к серверу и проверки корректности его ответов, а также для автоматизации тестирования API;

2 DBeaver – будет использоваться для администрирования и работы с базой данных, а также для выполнения SQL-запросов.

3 WEB-ресурс draw.io – будет использоваться для разработки UML-диаграмм;

4 WEB-ресурс drawSQL – будет использоваться для создания модели данных;

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		12

5 WEB-ресурс Figma – будет использоваться для создания UX и UI прототипов;

6 Microsoft Office Word – будет использоваться для написания документации к программному продукту;

7 Microsoft Office Visio – будет использоваться для создания диаграммы Ганта.

Разработка проекта будет происходить на компьютере со следующими параметрами:

- процессор 12th Gen Intel(R) Core(TM) i5-12450H 2.00 GHz;
- объем оперативной памяти 16.00 GB;
- объем места на жестком диске 460 GB;
- видеокарта Intel UHD Graphics;
- ОС Windows 11 Домашняя.

1.5 Разработка плана работы над проектом

Для эффективного управления процессом разработки и обеспечения своевременного выполнения проекта был составлен план работ, визуализированный с помощью диаграммы Ганта.

Диаграмма Ганта — это инструмент планирования проектов, представляющий собой столбчатую диаграмму, которая отображает задачи проекта, их продолжительность и взаимосвязи во времени. Данная диаграмма позволяет наглядно представить график работ, определить критический путь проекта и контролировать выполнение задач.

Диаграмма Ганта по разработке данного программного обеспечения представлена в Приложении Б.

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		13

2 Проектирование задачи

2.1 Разработка структуры сайта

Проектирование сайта — критически важный этап, определяющий удобство взаимодействия пользователей и эффективность ресурса. Ключевые аспекты — простота использования, удобная навигация, высокая скорость загрузки и соответствие сайта потребностям целевой аудитории — покупателей мужских штанов. Сайт должен быть интуитивно понятным и удобным для пользователей всех возрастов, желающих заказать штаны онлайн с возможностью добавления принта. Дизайн выполнен в коричнево-бежевых, белых, серых и черных тонах. Все основные разделы: «Главная», «Каталог», «Корзина» и «О нас» — легко доступны с главной страницы.

Полную структуру сайта, отражающую описанные принципы, можно рассмотреть в Приложении В.

2.2 Разработка UML-диаграмм

Диаграмма последовательности — это диаграмма, которая показывает взаимодействие объектов во времени, иллюстрируя последовательность передачи сообщений между объектами для выполнения определенного сценария. На данной диаграмме отображена последовательность действий гостя при просмотре каталога товаров и добавлении товара в корзину, включая взаимодействие с каталогом, корзиной и базой данных.

Диаграмму последовательности можно рассмотреть в приложении Г.

Диаграмма классов — это статическое представление структуры системы, описывающее классы, их атрибуты, методы и взаимосвязи. На данной диаграмме представлены классы «Категория», «Товар», «Подробности заказа», «Заказ» и «Администратор» с их атрибутами и методами, а также связи между классами.

Диаграмму классов можно рассмотреть в приложении Д.

Модель данных — это схема базы данных, представленная в форме диаграммы сущность-связь (ERD). На данной диаграмме показаны сущности «Категория», «Товар», «Подробности заказа», «Заказ» и «Администратор», их атрибуты и связи между ними.

Модель данных можно рассмотреть в приложении Е.

2.3 Разработка пользовательского интерфейса

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		14

Важным аспектом проектирования является разработка удобного и эффективного пользовательского интерфейса интернет-магазина. Для достижения этой цели были использованы сдержанные, приятные для восприятия цвета (коричневый, бежевый, белый, серый, черный), способствующие комфортному взаимодействию с сайтом.

В процессе проектирования особое внимание уделялось созданию целостного и интуитивно понятного интерфейса. Все компоненты интерфейса логически сгруппированы по функциональному назначению, что упрощает навигацию и позволяет пользователю быстро находить нужную информацию и выполнять целевые действия (просмотр товаров, добавление в корзину, оформление заказа). Благодаря такому подходу, взаимодействие с сайтом становится более эффективным и требует меньше усилий со стороны пользователя.

Разработка интерфейса началась с создания прототипов. Прототип – это экспериментальная модель интерфейса, создаваемая для тестирования и оценки функциональности, удобства использования и дизайна на ранних стадиях разработки. Он служит средством визуализации и проверки проектных решений, позволяя выявить потенциальные проблемы и внести необходимые корректировки до начала финальной разработки. Итоговый интерфейс может отличаться от прототипа.

Прототипы UX, представленные в Приложении И, фокусируются на логике взаимодействия пользователя с сайтом, тогда как прототипы UI, представленные в Приложении К, демонстрируют визуальное оформление сайта.

Со всеми разработанными UX прототипами можно ознакомиться по ссылке: <https://www.figma.com/design/4Y7LXlbG0yXqSlaeDnhYVs/UX>.

Со всеми разработанными UI прототипами можно ознакомиться по ссылке: <https://www.figma.com/design/AOgn7S6vbj5nAQ0E4LqNBh/UI>.

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
						15
Изм.	Лист	№ докцм.	Подпись	Дата		

3 Реализация

3.1 Руководство программиста

Данный интернет-магазин разработан с использованием современных веб-технологий, обеспечивающих высокую производительность, надежность и масштабируемость. В качестве основы для клиентской части (фронтенда) были выбраны HTML, CSS и JavaScript, в то время как серверная часть (бэкенд) реализована на Java с применением фреймворка Spring. Хранение данных осуществляется с помощью реляционной базы данных PostgreSQL. Для разработки использовались интегрированные среды разработки VS Code и IntelliJ IDEA, а также специализированные инструменты Postman и DBeaver для тестирования API и управления базой данных соответственно.

3.2 Создание сайта

3.2.1 Создание клиентской части

Разработка интернет-магазина началась с реализации фронтенда. Структура каждой веб-страницы была создана с использованием языка разметки HTML. Применение семантических тегов HTML5, таких как <header>, <nav>, <main> и <footer>, обеспечило не только логическую структуру документа, но и способствовало повышению доступности сайта для пользователей с ограниченными возможностями и улучшению SEO-оптимизации. Для отображения контента и создания интерактивных элементов использовались стандартные HTML-элементы, включая <div>, , , , , <input> и <button>. Эта тщательно продуманная разметка сформировала надежный фундамент для последующего стилизования и добавления динамического поведения. Стилизация пользовательского интерфейса была выполнена с помощью каскадных таблиц стилей (CSS). CSS позволил точно определить визуальное представление каждого элемента, включая цвета, шрифты, размеры, отступы, расположение и другие визуальные характеристики, обеспечивая единообразный дизайн и повышая удобство использования (юзабилити). Динамическое поведение и интерактивность были реализованы с помощью JavaScript. JavaScript позволил обрабатывать действия пользователей, такие как добавление товаров в корзину, взаимодействие с модальными окнами (например, для выбора размера, количества и загрузки изображения для принта). Разработка фронтенда велась в среде VS Code, которая, благодаря своей гибкости и расширяемости, обеспечила

					УП ТРПО 2-40 01 01.33.4.1.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		16

удобную работу с HTML, CSS и JavaScript файлами, а также интеграцию с системами контроля версий, такими как Git.

3.2.2 Создание серверной части

Серверная логика приложения (бэкенд) разработана на языке программирования Java с использованием фреймворка Spring. Spring обеспечил инфраструктуру для обработки HTTP-запросов от клиентской части, управления базой данных PostgreSQL и организации бизнес-логики приложения. Spring, как мощный и широко используемый фреймворк, значительно упростил разработку бэкенда благодаря своим возможностям внедрения зависимостей, аспектно-ориентированного программирования и управления транзакциями. В качестве системы управления базами данных (СУБД) была выбрана PostgreSQL, известная своей надежностью, гибкостью, расширяемостью и полной поддержкой стандарта SQL. Схема базы данных была спроектирована с учетом требований к эффективности хранения и извлечения данных о товарах, категориях, пользователях и заказах. Для упрощения взаимодействия с СУБД PostgreSQL и минимизации объема шаблонного кода был использован Spring Data JPA, предоставляющий удобную высокоуровневую абстракцию для работы с данными. Разработка бэкенда проводилась в среде IntelliJ IDEA, которая предлагает широкий набор инструментов для разработки на Java, включая интеграцию с Spring, мощные возможности отладки и профилирования кода, а также поддержку различных систем сборки и развертывания. Для управления базой данных PostgreSQL и выполнения SQL-запросов использовался DBeaver, кроссплатформенный SQL-клиент и инструмент администрирования баз данных. DBeaver предоставляет интуитивно понятный интерфейс для работы с базой данных, позволяя просматривать таблицы, выполнять запросы, экспортировать и импортировать данные. Для тестирования API бэкенда использовался Postman. Postman позволяет отправлять HTTP-запросы различных типов (GET, POST, PUT, DELETE и др.) к серверу, просматривать и анализировать полученные ответы, добавлять заголовки и параметры запросов, а также создавать коллекции запросов для автоматизации тестирования.

Интеграция фронтенда и бэкенда была осуществлена для обеспечения их взаимодействия. JavaScript на клиентской стороне отправляет запросы к серверу для получения и отправки данных. Эти запросы, как правило, выполняются асинхронно, чтобы не блокировать пользовательский интерфейс. Сервер, разработанный на Java с использованием Spring, обрабатывает эти запросы и

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		17

возвращает данные в формате JSON — удобном для обработки на клиентской стороне с помощью JavaScript. Данная архитектура, основанная на четком разделении ответственности между клиентом и сервером, позволила создать гибкое и масштабируемое приложение. Использованный стек технологий — HTML, CSS, JavaScript, Java, Spring и PostgreSQL — обеспечил эффективную реализацию поставленных задач и создание современного интернет-магазина. Postman использовался для тестирования и отладки взаимодействия между фронтендом и бэкендом, обеспечивая корректную работу всей системы.

Листинг приложения находится в приложении Л.

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		18

4 Тестирование

4.1 Тесты на использование

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Разработанные тест-кейсы и статус их выполнения представлены в приложение Ж.

Расписание работ над проектом представлено в таблице 5.

Таблица 5 – Расписание работ над проектом

Имя	Дата	Деятельность	Продолжительность, ч
Фалейчик Максим	11.12.2024	Разработка тестов	2
Фалейчик Максим	12.12.2024	Тестирование web-сайта	3
Фалейчик Максим	13.11.2024	Составление отчетов о найденных дефектах	3
Фалейчик Максим	14.11.2024	Исправление найденных ошибок	1
Фалейчик Максим	15.11.2024	Проведение регрессионного тестирования	2
Фалейчик Максим	16.11.2024	Составление отчета о результатах тестирования	3

4.2 Отчёт о результатах тестирования

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

Статистика по всем дефектам представлена в таблице 6.

Таблица 6 – Статистика по всем дефектам

Статус	Количество	Важность			
		Низкая	Средняя	Высокая	Критическая
Найдено	0	0	0	0	0
Исправлено	0	0	0	0	0
Проверено	0	0	0	0	0
Открыто заново	0	0	0	0	0
Отклонено	0	0	0	0	0

5 Руководство пользователя

Целью данного проекта является создание удобного и функционального интернет-магазина для продажи мужских штанов с возможностью персонализации за счет добавления принта. Данное руководство содержит информацию о том, как пользоваться сайтом, находить нужные товары, добавлять их в корзину, оформлять заказы и использовать другие функции магазина.

Для доступа к интернет-магазину перейдите по ссылке.

После перехода по ссылке вы будете перенаправлены на главную страницу сайта. На главной странице отображается основная информация о магазине, а также навигационные элементы для быстрого доступа к различным разделам сайта. Главная страница представлена на рисунке 1.

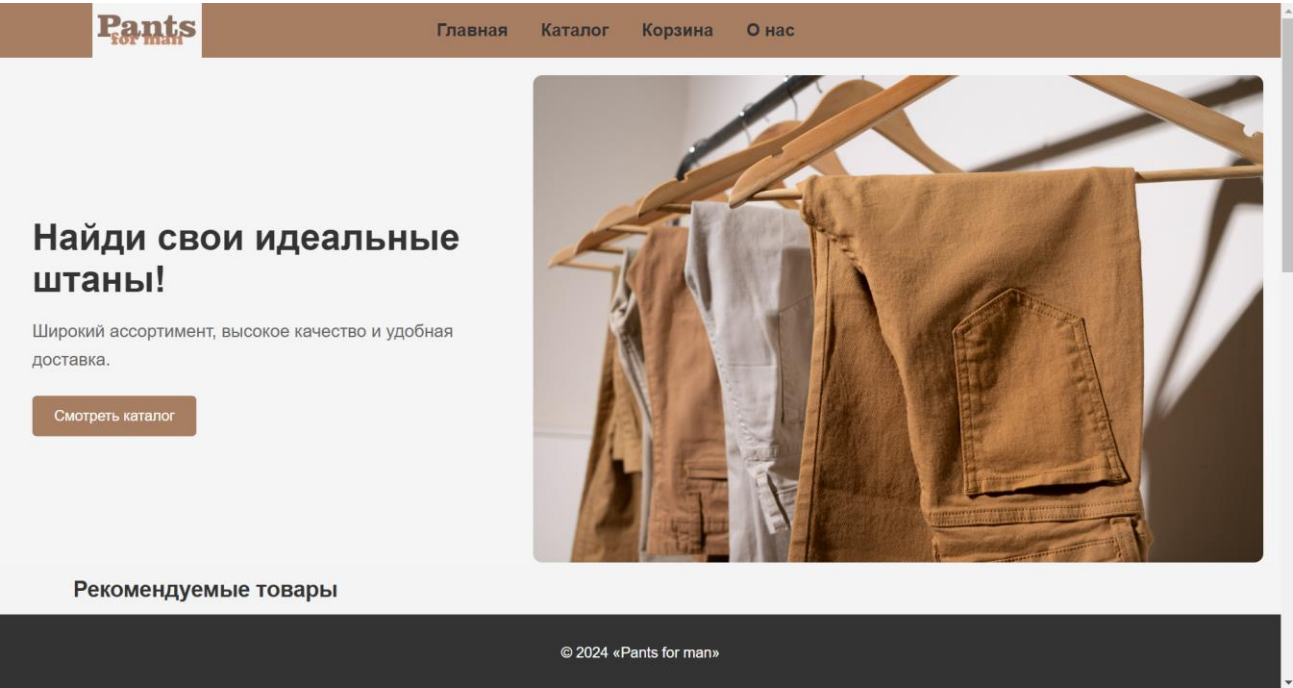


Рисунок 1 – Главная страница сайта

Затем на главной странице представлены рекомендуемые товары. Это позволяет посетителям сразу ознакомиться с подборкой интересных моделей штанов, которые могут привлечь их внимание. Рекомендуемые товары можно посмотреть на рисунке 2.

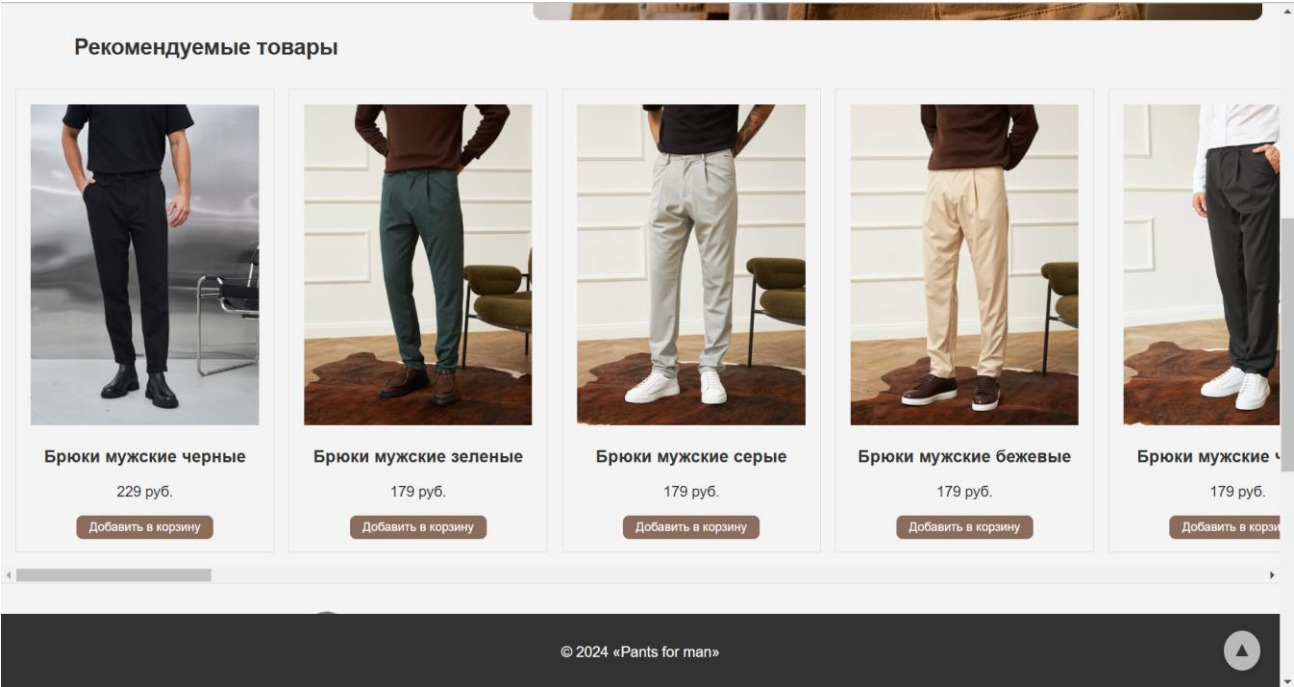


Рисунок 2 – Рекомендуемые товары

Также на главной странице находятся ссылки на основные категории товаров: джинсы, шорты, классические брюки. Это позволяет быстро перейти к просмотру товаров интересующей вас категории. Категории представлены на рисунке 3.

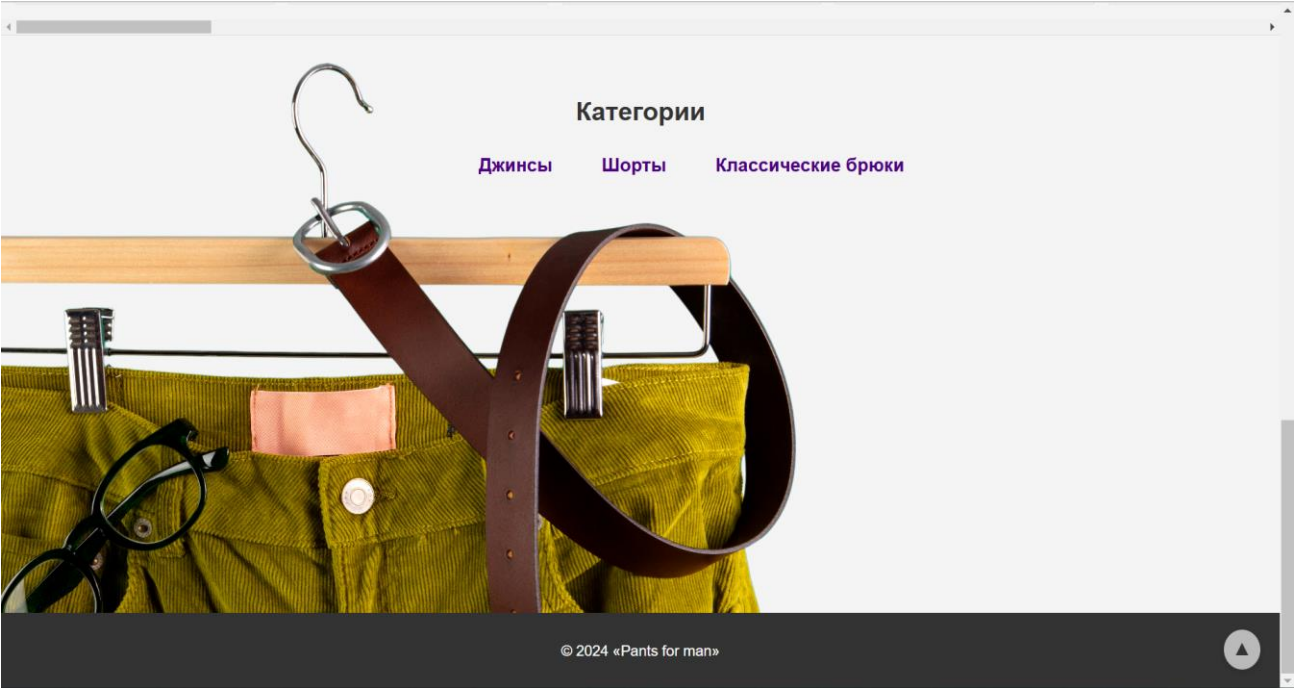


Рисунок 3 – Категории товаров

Кликнув на название категории, вы перейдете на страницу товаров этой категории. Например, на рисунке 4 представлена страница «Джинсы». На странице каталога товары представлены в виде карточек с фотографиями, названиями и ценами. Для удобства поиска можно использовать строку поиска, расположенную в верхней части страницы.

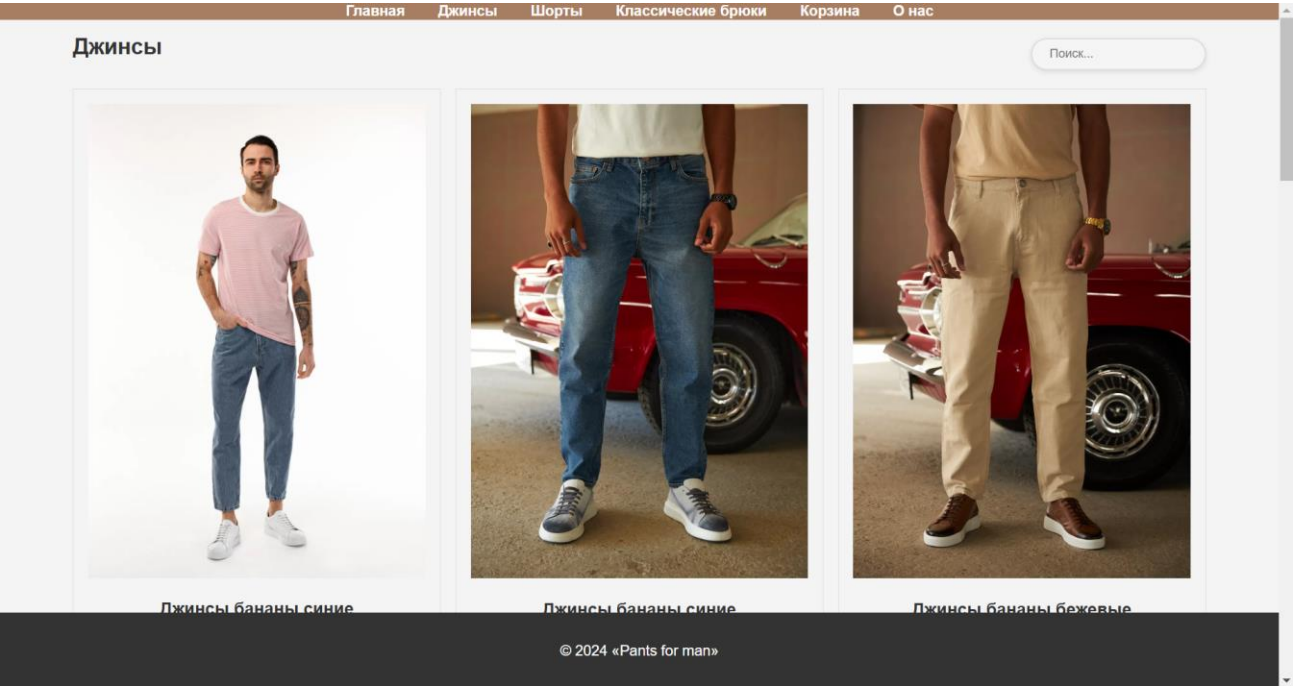


Рисунок 4 – Страница «Джинсы»

Чтобы добавить товар в корзину, нажмите кнопку «Добавить в корзину» на странице товара. Откроется модальное окно, изображенное на рисунке 5. В этом окне вы сможете выбрать размер и количество товара, а также загрузить изображение для нанесения принта на штаны. Обратите внимание на требования к формату и размеру изображения, которые указаны в модальном окне. Перед добавлением товара в корзину вы сможете предварительно просмотреть загруженное изображение, чтобы убедиться, что оно соответствует вашим ожиданиям.

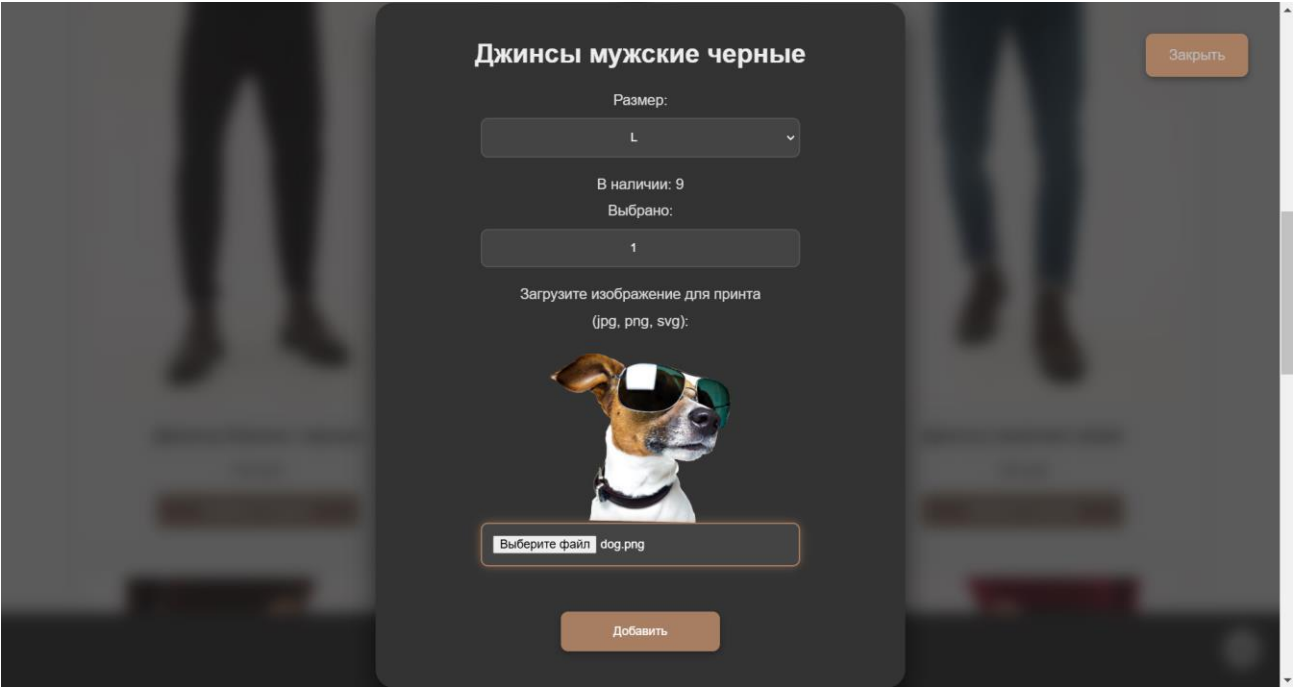


Рисунок 5 – Модальное окно добавления в корзину

После добавления всех необходимых товаров в корзину перейдите в раздел «Корзина», изображенный на рисунке 6. В корзине вы увидите список всех выбранных товаров, их количество, размеры и общую стоимость заказа. Здесь вы можете изменить количество товаров, удалить ненужные позиции или продолжить оформление заказа.

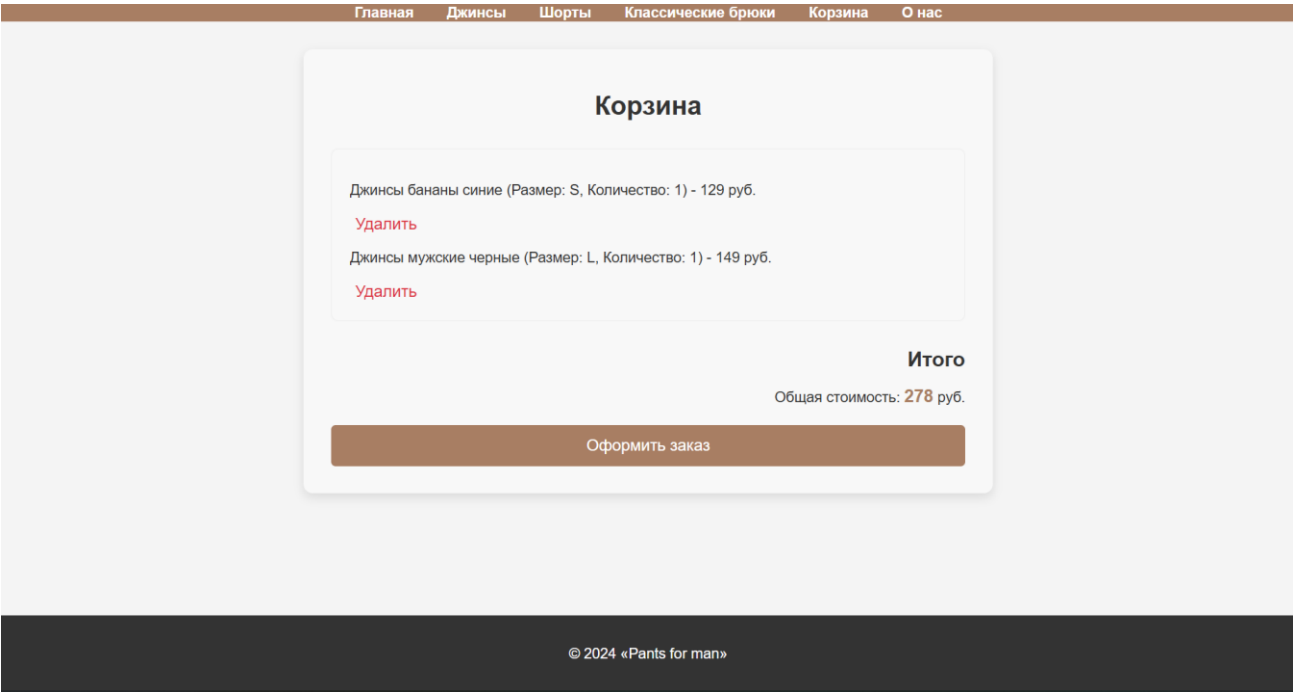


Рисунок 6 – Страница «Корзина»

Для оформления заказа нажмите кнопку «Оформить заказ». После нажатия на эту кнопку откроется модальное окно «Информация о заказе», представленное на рисунке 7, где необходимо заполнить все необходимые поля: контактная информация, адрес доставки, и номер телефона.

Рисунок 7 – Модальное окно «Информация о заказе»

После заполнения всех строк в модальном окне «Информация о заказе» и нажатия кнопки «Подтвердить заказ» будет выведено сообщение об успешном оформлении заказа. Это сообщение показано на рисунке 8.

Рисунок 8 – Сообщение об успешном оформлении заказа

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		24

Заключение

Целью данного проекта являлось создание удобного и функционального интернет-магазина для продажи мужских штанов с возможностью персонализации товаров путем добавления принта. В ходе работы над проектом были разработаны все необходимые компоненты, включая главную страницу с презентацией рекомендуемых товаров и категорий, каталог товаров с возможностью поиска по заголовкам товаров, функционал корзины и модальное окно для добавления товара с опцией загрузки принта. Также была реализована форма оформления.

В процессе разработки были использованы современные технологии веб-разработки, обеспечивающие высокую производительность и надежность работы интернет-магазина. Особое внимание уделялось созданию интуитивно понятного и привлекательного пользовательского интерфейса. Также были закреплены знания по предмету «Технология разработки программного обеспечения».

Проект был тщательно протестирован на разных этапах разработки, что позволило выявить и исправить все ошибки и недочеты. В результате был создан полностью функциональный интернет-магазин, соответствующий поставленным требованиям и готовый к использованию. Успешная реализация проекта подтверждает эффективность выбранных методов и технологий разработки.

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		25

Список использованных источников

1 Принципы SOLID на примерах [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/articles/688530> – Дата доступа: 01.12.2024.

2 Трехслойная архитектура [Электронный ресурс]. – Режим доступа: <https://doka-guide.vercel.app/tools/clean-architecture> – Дата доступа: 27.11.2024.

3 Форум программистов и сисадминов Киберфорум [Электронный ресурс]. – Режим доступа: <https://www.cyberforum.ru> – Дата доступа: 10.12.2024.

4 CSS Tutorial [Электронный ресурс]. – Режим доступа: <https://www.w3schools.com/css> – Дата доступа: 15.11.2024.

5 Клуб ПРОграммистов [Электронный ресурс]. – Режим доступа: <https://programmersforum.ru> – Дата доступа: 11.12.2024.

					УП ТРПО 2-40 01 01.33.41.21.24 ПЗ	Лист
Изм.	Лист	№ докцм.	Подпись	Дата		26