

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ»
(ГАПОУ СПО ОКЭИ)

КУРСОВОЙ ПРОЕКТ

ОКЭИ 09.02.07. 4323. № 26 ПЗ
(код документа)

Количество листов 30
Дата готовности 28.12.2023
Руководитель Лукасян А.Д.
Разработал Широнин Н.С.
Защищен _____ с оценкой _____
(дата)

Оренбург 2023

Содержание

| | |
|--|----|
| Введение | 3 |
| 1 Анализ предметной области | 5 |
| 2 Проектирование приложения | 8 |
| 3 Разработка программного обеспечения | 10 |
| 3.1 Описание технологического стека разработки | 10 |
| 3.2 Описание алгоритма работы | 12 |
| 3.3 Описание интерфейса пользователя | 13 |
| 4 Тестирование приложения | 14 |
| 4.1 План тестирования | 14 |
| 4.2 Оценка результатов проведения тестирования | 16 |
| Заключение | 22 |
| Список используемых источников | 23 |
| Приложение А (обязательное) | 25 |
| Приложение Б (обязательное) | 26 |
| Приложение В (обязательное) | 27 |
| Приложение Г (обязательное) | 28 |
| Приложение Д (обязательное) | 30 |

| | | | | | | | | | | | |
|-----------|------|--------------|---------|----------------------------|--|--|--|------|------|--------|--|
| | | | | | ОКЭИ 09.02.07. 9023 1 П | | | | | | |
| | | | | | | | | | | | |
| Изм. | Лист | № докум. | Подпись | Дата | | | | | | | |
| Разраб. | | Широнин Н.С. | | | Отчёт по практике по профилю специальности | | | Лит. | Лист | Листов | |
| Провер. | | | | | | | | | 2 | 59 | |
| Реценз | | | | Отделение — очное гр. 4бд1 | | | | | | | |
| Н. Контр. | | | | | | | | | | | |
| Утверд. | | | | | | | | | | | |

Введение

Веб-приложение – это программное обеспечение, разработанное специально для работы в сети Интернет. Оно доступно пользователю через веб-браузер и предоставляет функциональность, которая включает в себя обработку данных, визуализацию информации, взаимодействие с базой данных, отправку и получение запросов и многое другое.

Они могут легко включать в себя инструменты для анализа данных и создания отчетов, что помогает в принятии информированных решений.

Пользователи могут использовать веб-приложения на разных устройствах, будь то компьютеры, смартфоны, планшеты или даже смарт-телевизоры, так как они не зависят от операционной системы.

Таким образом, веб-приложения широко используются в бизнесе, образовании, здравоохранении, государственном управлении, развлечениях и многих других сферах для улучшения производительности, доступности к информации и взаимодействия с пользователями.

В ходе курсовой должно быть решено несколько ключевых задач:

- разработка серверной части веб-приложения, включая создание интерфейса и определение функциональности;
- разработка системы.

Целью создания такого веб-приложения является улучшение доступности и удобства обслуживания клиентов.

Для успешной реализации намеченных задач требуется провести комплексную работу, которая включает в себя несколько ключевых этапов:

Прежде всего, необходимо определить и разработать логическую структуру будущего веб-приложения. Это включает в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, каким образом приложение будет собирать, хранить и обрабатывать.

Для успешной реализации намеченных задач требуется провести комплексную работу, которая включает в себя несколько ключевых этапов:

- определить и разработать логическую структуру будущего веб-приложения;

Это включает в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, как пользователи будут взаимодействовать с системой.

- для удобства использования веб-приложения необходимо разработать дизайн пользовательского интерфейса;

Это включает в себя выбор цветовой палитры, шрифтов, структуры страниц, расположения элементов управления и визуального оформления. Для реализации данного пункта должны быть разработаны эскизы и макеты будущего веб-приложения.

- после разработки дизайна необходимо перейти к вёрстке страниц приложения;

Этот этап включает в себя создание HTML, CSS, JS, Bootstrap кода для каждой страницы, определённой в логике приложения. Важно обеспечить адаптивность и кросс-браузерность, чтобы приложение корректно отображалось на различных устройствах и в разных браузерах.

- разработать архитектуру базы данных для разрабатываемой серверной части, включая выбор подходящих таблиц;

- завершив этап разработки архитектуры базы данных, перейдем к созданию серверной части.

Этот этап включает в себя подключение библиотеки ReadBeanPHP (ORM).

Для успешного решения этой проблемы необходимо тщательно спроектировать архитектуру приложения, уделяя особое внимание взаимосвязям между его компонентами, эффективности обработки данных и запросов пользователей, а также обеспечить масштабируемость при увеличении нагрузки.

Этот комплексный подход позволит создать полноценное приложение, которое удовлетворит потребности пользователей и поможет в достижении поставленных целей.

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 4 |

1 Анализ предметной области

NBB интернет магазин включает в себя услуги электронной коммерции. электронику, косметику, товары для дома и многое другое. Наш интернет-магазин позволяет клиентам удобно и безопасно совершать покупки, делать заказы онлайн и получать доставку в удобное для них время и место. Мы также предоставляем услуги по обработке платежей и обслуживанию клиентов, чтобы обеспечить нашим клиентам наилучший опыт покупок. В предметной области интернет-магазина проводится анализ, который помогает определить основные характеристики и требования к магазину. Вот несколько основных направлений анализа предметной области:

- рынок и конкуренция (исследование рынка помогает определить долю рынка, понять потребности и предпочтения целевой аудитории, а также выявить конкурентов и изучить их стратегии);
- товары и категории (анализ товаров и категорий позволяет выявить самые востребованные товары и установить оптимальную структуру каталога);
- цены и акции (анализ цен и акций помогает определить оптимальные ценовые диапазоны, стратегии скидок и акций для привлечения клиентов);
- корзина и оформление заказа (анализ корзины и процесса оформления заказа помогает определить факторы, влияющие на конверсию клиентов и улучшить пользовательский опыт);
- логистика и доставка (анализ логистики и доставки позволяет оптимизировать процесс доставки товаров, выбрать наилучшие партнеры и улучшить качество обслуживания клиентов).

Все эти аспекты анализируются с целью оптимизации и улучшения работы интернет-магазина, удовлетворения потребностей клиентов и достижения успеха в данной предметной области.

Прежде чем начать проектировать веб-приложение «NBB», надо посмотреть структуру аналогичных веб-приложений, т.к. их разработчики разгадали «черный ящик» и выявили работающие гипотезы, другими словами, выполнили всю работу за нас.

Проводя оптимизационные работы таким методом, разработчик совершает намного меньше ошибок. Кроме того, отсутствует необходимость проверять и перепроверять огромное количество различных гипотез. Как показывает практика, копирование и внедрение «фишек» других сайтов на своем ресурсе дает максимально быстрый рост позиций.

В интернете множество сайтов с электроникой например сайт <https://www.dns-shop.ru/> сайт имеет приятный дизайн у него удобный пользовательский интерфейс все товары отсортированы по разделам и категориям есть функция корзины для выбранных товаров.

Прежде чем приступить к созданию Web-приложения, необходимо определить его функциональные возможности.

Функционал сайта - это то, что делает веб-ресурс полезным, что превращает его из обычного источника информации в полезную программу. Создание сложных сайтов, таких как Интернет-каталоги невозможно без разработки специальных программных модулей.

Программные модули сделают сайт интересным, удобным и полезным. Различные программные модули позволят сайту гостеприимно встречать своих гостей, узнавать своих гостей, выяснять их интересы и предпочтения, оперативно отвечать на их вопросы, помогать быстро и легко найти необходимые вещи, данные. Системы поиска по сайту, каталог одежды, блоки анонсов разделов помогут быстро найти необходимое..

Во время разработки Web-приложения каталога магазина электротехники необходимо учесть, что дизайн всех страниц сайта должен быть выполнен в единой цветовой гамме и едином стиле, а также он должен зависеть от функциональных возможностей и удобства использования сайта. В верхней части всех страниц сайта должен быть расположен логотип сайта.

Шрифты в текстах должны быть легко читаемыми. У пользователя должна быть возможность регистрации на сайте, т.к. без нее он не сможет проголосовать за тот или иной препарат и участвовать в обсуждениях. На сайте должна присутствовать только значимая для клиентов информация о компании, способах связи и товарах. Для улучшения навигации по сайту должно быть разработано меню в привычном и интуитивно понятном пользователям составе.

Главным моментом является то, что у покупателя обязательно должна быть возможность купить товар но так как это каталог а не магазин то нужно дать очень конкретные ссылки на магазины в которых существует данный товар . Для этого должна присутствовать так называемая «Корзина», которая должна быть отображена на всех страницах сайта.

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 6 |

2 Проектирование приложения

Проектирование интернет-магазина включает в себя несколько основных этапов:

- анализ требований и определение целей (на этом этапе необходимо провести анализ рынка, изучить потребности целевой аудитории, определить основные цели и задачи интернет-магазина);

- проектирование структуры сайта (следующим шагом является создание структуры сайта, которая определяет навигацию, категории товаров, страницы продуктов и другие разделы);

- разработка дизайна (на этом этапе создается дизайн интерфейса сайта с учетом бренд-идентичности, эстетических предпочтений аудитории и удобства использования. Важно учесть адаптивность дизайна для удобного просмотра на различных устройствах);

- разработка функционала (затем необходимо определить функциональные возможности интернет-магазина, такие как поиск и фильтрация товаров, корзина покупок, оформление заказа, управление аккаунтом и др. Важно убедиться, что все функции работают правильно и без ошибок);

- разработка базы данных (для хранения информации о товарах, клиентах, заказах и других данных требуется создать базу данных и разработать соответствующую структуру);

- тестирование и отладка (перед запуском сайта необходимо провести тестирование всех функций и проверить их работоспособность. Ошибки и неполадки должны быть исправлены, чтобы обеспечить безупречную работу интернет-магазина);

- запуск и поддержка (после успешного завершения всех предыдущих этапов интернет-магазин готов к запуску. Однако поддержка и обновление магазина должны осуществляться на постоянной основе, чтобы удерживать его в актуальном состоянии и обеспечивать удобство пользования для клиентов).

Функционал интернет-магазина может включать в себя следующие возможности:

- поиск и фильтрация товаров (пользователь может искать и отфильтровывать товары по разным критериям, таким как цена, бренд, тип товара и т.д.);

- корзина покупок (пользователь может добавлять товары в корзину и управлять содержимым, включая изменение количества, удаление товаров и расчет общей стоимости);

- оформление заказа (пользователь может выбирать тип доставки, указывать адрес доставки и оплачивать товары);

- отзывы и рейтинги товаров (пользователь может оставлять отзывы о приобретенных товарах и оценивать их качество, что помогает другим покупателям принять решение);

- интеграция с платежными системами (интернет-магазин может

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 7 |

поддерживать различные способы оплаты, такие как кредитные карты, электронные платежи и другие);

– уведомления и обратная связь (пользователь может получать уведомления о статусе заказа, изменении цен и других событиях. Также магазин может предоставлять возможность связаться со службой поддержки через онлайн-чат, форму обратной связи или телефон);

– подборки и рекомендации (на основе предпочтений и истории покупок пользователя магазин может предлагать персонализированные подборки товаров и рекомендации);

– программа лояльности (интернет-магазин может предлагать бонусы, скидки и специальные предложения своим постоянным покупателям).

Функции пользователя:

- просмотр товара;
- добавления товара в корзину;
- удаление товара;
- фильтры товара;
- заказ товара.

Функции Администратора:

- просмотр статистики;
- просмотр заказов;
- добавления товара;
- удаления товара;
- управления учётными записями пользователей.

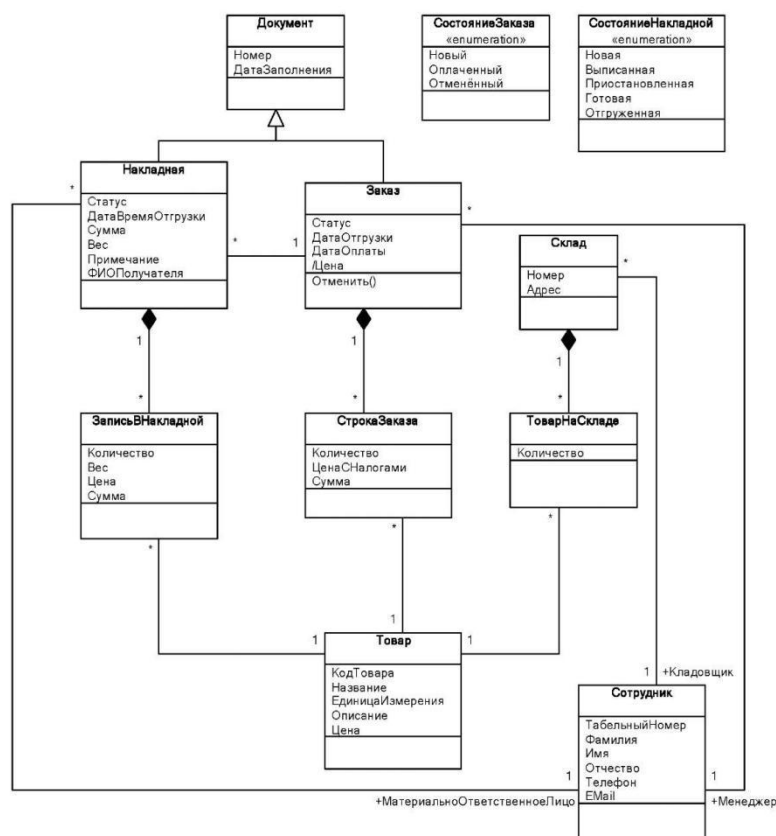


Рисунок 1 – Схема базы данных

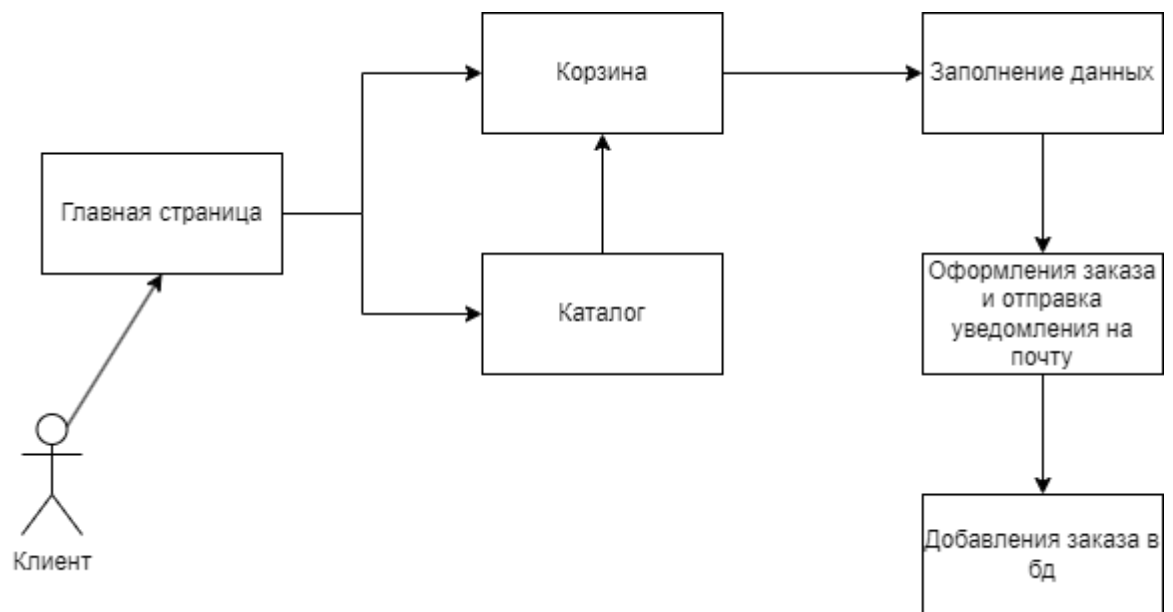


Рисунок 2 – Схема работы интернет магазина

3 Разработка программного обеспечения

3.1 Описание технологического стека разработки

3.1.1 Среда разработки

В качестве интегрированной среды разработки рекомендуется использовать Visual Studio Code или PhpStorm. Для удобства работы с IDE могут использоваться такие плагины как Emmet.

3.1.2 Удаленный доступ

Для удаленного управления и развертывания приложения рекомендуется использовать SSH.

3.1.3 Технологии

Для реализации веб-приложения рекомендуется использовать веб-сервер Nginx или Apache.

В качестве языков программирования должны быть использованы:

PHP 8.0 для разработки серверной логики;

HTML для написания структуры сайта;

JavaScript для клиентской логики.

CSS для создания и изменения стиля элементов веб-страниц и пользовательских интерфейсов.

Для реализации некоторых функций могут быть использованы библиотеки и фреймворки:

PDO для упрощения работы с БД;

Bootstrap для быстрого написания CSS кода.

3.2 Описание алгоритма работы

Алгоритм работы интернет-магазина может выглядеть следующим образом:

- пользователь заходит на сайт интернет-магазина;
- пользователь может воспользоваться функцией поиска для быстрого нахождения нужного товара. Также можно использовать фильтры для уточнения результатов поиска;
- пользователь выбирает интересующий его товар и переходит на страницу с подробной информацией о товаре. Здесь пользователь может ознакомиться с описанием, характеристиками, ценой, наличием и отзывами о товаре;
- если товар удовлетворяет требованиям покупателя, он может добавить его в корзину;
- в корзине пользователь может изменять количество товаров, удалять ненужные товары или продолжить покупки, добавив еще товары в корзину;
- когда пользователь закончил выбор товаров, он переходит к оформлению заказа. Здесь ему предлагается заполнить форму с контактной информацией и выбрать способ доставки и оплаты.

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | - | 10 |

- после заполнения формы данных и выбора всех деталей заказа, пользователь подтверждает заказ;
- после подтверждения заказа пользователю предоставляется информация о деталях заказа, и он может произвести оплату;
- когда оплата прошла успешно, интернет-магазин отправляет уведомление о заказе на электронную почту покупателя и начинает обработку заказа;
- заказ обрабатывается сотрудниками магазина. Товары подготавливаются к отправке, упаковываются и передаются службе доставки;
- покупатель получает заказ по выбранному им способу доставки;
- после получения товаров пользователь может оставить отзыв о покупке и оценить качество обслуживания.

3.3 Описание интерфейса пользователя

Интерфейс пользователя интернет-магазина электротехники должен быть интуитивно понятным, удобным и легко наведируемым. Вот некоторые основные элементы, которые могут входить в состав интерфейса:

Шапка сайта. В верхней части страницы располагается шапка сайта, где обычно размещается логотип, основное меню навигации, поиск по сайту и ссылки на дополнительную информацию (контакты, о нас и т.д.).

Основное меню навигации. В основном меню навигации представлены основные категории товаров, такие как бытовая техника, электроника, компьютеры и т.д. При наведении курсора на каждую категорию, может открываться выпадающее подменю с более подробными подкатегориями.

Блоки с акциями и новинками. На главной странице или на отдельной странице могут быть размещены блоки с акционными предложениями, скидками, новинками и самыми популярными товарами. Это позволяет привлечь внимание пользователей к актуальным предложениям.

Фильтры и сортировка. Для удобства поиска товаров пользователю должны быть предоставлены фильтры по характеристикам товаров (например, цена, бренд, тип, мощность и т.д.), а также опции сортировки результатов поиска (например, по цене, популярности, новизне и т.д.).

Каталог товаров. Обычно он расположен в левом сайдбаре (левой части сайта) или по центру выше середины экрана. В зависимости от масштабов магазина, отличаться будет и каталог товаров: его структура, размер и расположение. Например, крупный интернет-ритейлер Amazon имеет 2х-уровневый каталог товаров, во втором уровне также присутствует тематическая реклама. Если у вас небольшой магазин с 5-7 товарными группами – вы можете расположить каталог горизонтально, с выпадающим меню подкатегорий.

Центральная часть. По данным исследований, человек подсознательно запоминает левый верхний угол, область логотипа, которую мы описывали выше. Здесь нужно располагать информацию, которая сможет заинтересовать, вызвать интерес, заманить. А это – наиболее популярные товары, товары со

скидкой, новые поступления, хиты продаж. Также используйте продающие тексты акций, например: «Товар дня: Блендер. 599 грн. Вместо 900, осталось 5 штук». Одним словом, всё то, что сможет зацепить пользователя, заинтересовать его и убедить остаться. Здесь же расположен и промо-блок.

Промо-блок. Это область в верхней части страницы (ниже шапки сайта). Промо-блок виден сразу при попадании на главную страницу сайта, поэтому размещать здесь информацию очень эффективно. В промо-блоке обычно размещают рекламу товара или группы товаров, акционные предложения, скидки. Одним словом, здесь стимулируются продажи самых выгодных для магазина товаров.

Не стоит, однако, отклоняться от традиционного вертикального или горизонтального положения меню. Пользователи давно привыкли к такому расположению, не нужно пытаться привить им какой-то креатив, это не тот случай. Не располагайте меню в правой части экрана, это «слепая зона» для пользователей, все привыкли, что именно справа находится только реклама и нет ничего полезного. Найдя каталог на привычном месте, человек всегда без проблем доберется до страницы с интересующим его товаром.

Карточки товаров. Каждый товар должен иметь свою карточку с основной информацией о товаре, такую как название, описание, изображение, цена, наличие на складе, рейтинг и отзывы покупателей. Также в карточке товара может быть возможность добавить товар в корзину или в список желаемых товаров.

Корзина. Корзина позволяет пользователям добавлять товары, удалять их, изменять количество и оформлять покупки. В корзине должна быть отображена общая сумма заказа, а также информация о доставке и оплате.

Профиль пользователя. Зарегистрированным пользователям должен быть доступен функционал профиля, где они могут управлять своими персональными данными, адресами доставки, просматривать историю заказов и управлять списками желаемых товаров.

Оформление заказа. При оформлении заказа пользователю предлагается выбрать способ доставки, метод оплаты и ввести необходимые данные для доставки.

Контактная информация. Раздел с контактной информацией, где указаны данные для связи с магазином, а также форма обратной связи для отправки вопросов или запросов.

Блоки с дополнительной информацией. На сайте может быть размещена информация о гарантии, возврате товаров, условиях доставки, контактах и других полезных сведениях.

Общие рекомендации:

- использование современного и эстетичного дизайна, приятного для глаза и удобного для восприятия;
- тщательное тестирование интерфейса на различных устройствах и браузерах, чтобы обеспечить его отзывчивость и адаптивность;
- предоставление пользователям возможности оставлять отзывы, рейтинги и комментарии к товарам, что повысит доверие к магазину.

4 Тестирование приложения

4.1 План тестирования

Для тестирования веб-приложения был выбрано несколько методов тестирования.

Предварительное тестирование:

- проверка совместимости интерфейса с различными браузерами (Chrome, Firefox, Safari, Edge и т. д.);
- проверка отзывчивого дизайна и адаптивности интерфейса на различных устройствах (компьютеры, планшеты, мобильные телефоны);
- проверка работы интерфейса на разных операционных системах (Windows, macOS, Android, iOS).

Тестирование функциональности:

- регистрация и авторизация пользователей;
- поиск и фильтрация товаров;
- просмотр детальной информации о товаре и добавление его в корзину;
- расчет общей стоимости заказа, применение скидок и промо-кодов;
- оформление заказа и оплата товаров;
- управление профилем пользователя, изменение персональных данных и адресов доставки;
- добавление и удаление товаров из списка желаемых.

Тестирование корзины и оформления заказа:

- проверка добавления товаров в корзину и изменения их количества;
- проверка удаления товаров из корзины;
- проверка вычисления общей стоимости заказа и применения доставки и налогов;
- проверка выбора и просмотра доступных способов доставки и оплаты;
- проверка правильной обработки данных пользователя (адрес доставки, контактная информация).

Тестирование процесса оформления заказа:

- проверка заполнения обязательных полей при оформлении заказа;
- проверка правильности работы расвета промежуточных и итоговых сумм заказа;
- проверка отправки подтверждения заказа на почту пользователя;
- проверка правильности работы системы платежей и получения платежных уведомлений.

Тестирование безопасности:

- проверка защиты информации пользователей (SSL-сертификат, шифрование данных);
- проверка безопасности платежных операций и передачи финансовых данных;
- проверка защиты от несанкционированного доступа к личным данным

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | - | 13 |

пользователей.

Тестирование производительности:

- проверка скорости загрузки страниц, особенно на мобильных устройствах;
- проверка работы интерфейса при большом количестве одновременных пользователей;
- проверка работы при нагрузке на сервер и контрольные точки сети.

Тестирование на различных стадиях разработки и обновления:

- проверка корректности работы новых функций и изменений;
- проверка обратной совместимости с предыдущими версиями сайта;
- проверка наличия и работоспособности всех ссылок и переходов на другие страницы.

Тестирование системы обратной связи:

- проверка работоспособности формы обратной связи;
- проверка получения уведомлений о входящих сообщениях;
- проверка корректного отображения сообщений в административной панели.

Тестирование на разных языках:

- проверка локализации интерфейса на различные языки, включая русский, английский и т.д.;
- проверка корректного отображения и ввода символов для каждого языка.

Тестирование на мобильных устройствах:

- проверка отзывчивого дизайна и адаптивности интерфейса на различных мобильных устройствах;
- проверка корректности работы интерфейса на разных операционных системах (Android, iOS).

DDOS attack. Чат-бот может быть эффективным только в том случае, если он может быть устойчивым к DDOS атакам. Тестирование реализовано на рисунке 3.

Код программы тестирования DDOS атаки на веб-приложения представлен в приложении Д.

4.2 Оценка результатов проведения тестирования

```
ddosddos ~/Desktop $ sudo nmap -sV --script=default --script=vulscan/vulscan.nse 192.168.1.6 -p8080
Starting Nmap 7.81 ( https://nmap.org ) at 2017-08-21 17:31 ICT
Nmap scan report for metasploit3 (192.168.1.6)
Host: is up (0.0001s latency)
PORT      STATE SERVICE VERSION
8080/tcp   open  http      Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)
http-methods:
  Potentially risky methods: PUT DELETE TRACE
  http-open-proxy: Proxy might be redirecting requests
  http-server-header: GlassFish Server Open Source Edition 4.0
  http-title: GlassFish Server - Server Running
vulscan: scip VulDB - http://www.scip.ch/en/vuldb:
[103593] Oracle GlassFish Server Open Source Edition 4.1 HTTP GET Request directory traversal
[109159] Oracle Solaris Cluster 4.2 GlassFish Server unknown vulnerability
[100132] Oracle Retail Open Commerce Platform 4.0/5.0/5.1/5.3/6.0 Framework information disclosure
[82677] Oracle Solaris Cluster 4.2 GlassFish Server information disclosure
[80535] Oracle VM Virtualbox up to 4.0.30/4.1.0/4.2.2/36/4.2.34/5.0.10 Core buffer overflow
[80536] Oracle Retail 4.0/4.1 Order Broker Cloud Service unknown vulnerability
[78622] Oracle Communications: Tekelac HLR Router 4.0.0 unknown vulnerability
[74879] Oracle iPlanet Web Proxy Server 4.0 Network security service spoofing
[67060] Oracle iPlanet Web Proxy Server 4.0.24 Security denial of service
[11230] Oracle iPlanet Web Proxy Server 4.0 Administration unknown vulnerability
[11826] Oracle iPlanet Web Proxy Server 4.0 Security Out-of-Bounds denial of service
[15172] Oracle Database 4.0/4.1 Application Express unknown vulnerability
[59134] Oracle Virtualization 4.0 unknown vulnerability
[59122] Oracle Database Server 4.0 Application Express unknown vulnerability
[50150] Oracle SunMC 4.0 Web Console unknown vulnerability
[54071] Oracle Sun Java System Web Proxy Server 4.0.13 Administration Server unknown vulnerability
[19795] Oracle Application Server 4.0.8 PL/SQL Module Format String
[16934] Oracle Application Server 4.0.8.2 HTTP Request header buffer overflow
[15737] Oracle Web Listener 4.0.7.0/4.0.8.1.0 on AIX URL denial of service
[15392] Oracle Application Server 4.0 on Win NT Web Listener /ows-bin privilege escalation
[103800] Oracle Hospitality e7 4.2.1 information disclosure
[103977] Oracle Hospitality Guest Access 4.2.0/0/4.2.1.0 information disclosure
[103976] Oracle Hospitality Guest Access 4.2.0/0/4.2.1.0 unknown vulnerability
[103944] Oracle Communications Network Charging and Control 4.4.1.5/5.0.8/5.0.2/5.0.1.0/5.0.2.0 OpenSSL unknown vulnerability
```

Рисунок 3 – Тестирование DDOS атака

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 14 |

Заключение

В ходе курсовой работы были выполнены работы по проектированию и разработке веб-приложения.

Для реализации веб-приложения был проведен анализ предметной области, составлено техническое задание, в котором определялись аппаратные и технические требования к разработке, была выполнена разработка проектной документации, в том числе подготовлены макеты и эскизы веб-приложения, реализован подбор графических материалов, после чего была выполнена разработка веб-приложения, включающая в себя реализацию шаблонов и динамических элементов веб-приложения и выполнен контрольный пример и были реализованы следующие функции:

- просмотр товаров;
- добавления товара;
- удаление товара;
- добавления, редактирование товара администратором;
- просмотр статистики;
- автоматизировать обработку запросов;
- осуществлять поиск и сбор данных.

В ходе курсовой работы были решены следующие задачи:

- определена и разработана логическая структура будущего веб-приложения, что включало в себя определение основных функциональных модулей, спецификацию их задач и взаимодействия между собой, также необходимо продумать, как пользователи будут взаимодействовать с системой;

- для удобства использования веб-приложения необходимо разработать дизайн пользовательского интерфейса. Это включает в себя выбор цветовой палитры, шрифтов, структуры страниц, расположения элементов управления и визуального оформления. Для реализации данного пункта были разработаны эскизы и макеты будущего веб-приложения;

- после разработки дизайна было решено перейти к верстке страниц приложения, что включало в себя создание HTML, CSS, JS, Bootstrap кода для каждой страницы, определенной в логике приложения. Важно было обеспечить адаптивность и кросс-браузерность, чтобы приложение корректно отображалось на различных устройствах и в разных браузерах, что и было достигнуто;

- была произведена разработка архитектуры базы данных для разрабатываемой серверной части, включая выбор подходящих таблиц;

- завершив этап разработки архитектуры базы данных, был совершён переход к созданию серверной части. Этот этап включает в себя подключение библиотеки ReadBeanPHP (ORM) и написания API для веб-приложения.

Подводя итоги выпускной квалификационной работы, можно сделать вывод, что поставленная цель исследования была достигнута, качество обработки клиентских обращений стало выше, путём разработки и внедрения на хостинг в декабре 2023 года.

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 15 |

Список использованных источников

- 1 Алексеев Ю.М. Быстро и легко создаем, программируем, шлифуем и раскручиваем web-сайт: учеб. пособие / Ю.М. Алексеев. - М.: Лучшие книги, 2011. - 189 с.
- 2 Ахромов Я.В. Технологии Web-дизайна и Flash-технологии: учеб. пособие / Я.В. Ахромов. - Ростов н/Д: Феникс, 2009. - 265 с.
- 3 Басыров Р. 1С-Битрикс: постройте профессиональный сайт сами / Р. Басыров. - СПб.: Питер, 2010. - 237 с.
- 4 Белунцов В. Новейший самоучитель по созданию Web-страниц / В. Белунцов. - М.: NT Press, 2012. - 185 с.
- 5 Берд Дж. Веб-дизайн. Руководство разработчика: учебник. - СПб.: Питер, 2012. - 224 с.
- 6 Благодатских В.А. Стандартизация разработки программных средств: учеб. пособие / В.А. Благодатских. - М.: Финансы и статистика, 2011. - 544 с.
- 7 Бородаев Д.В. Web-сайт как объект графического дизайна: монография / Д.В. Бородаев. - Харьков: «Септима ЛТД», 2011. - 288 с.
- 8 Браун Брэдли Oracle Database. Создание Web-приложений: учебник / Брэдли Браун, Ричард Ниемик, Джозеф С. Треззо. - СПб.: Лори, 2011. - 722 с.
- 9 Смирнова И. Начала web-дизайна: учеб. пособие / И. Смирнова. - СПб.: БХВ-Петербург, 2009. - 231 с.
- 10 Советов Б.Я. Информационные технологии: учебник для ВУЗов / Б.Я. Советов, В.В. Цехановский. - М.: Высшая школа, 2010. - 263 с.
- 11 Стогний А.А. Основы компьютерной грамотности: учебник / А.А. Стогний. - Киев: Головное издательство, 2009. - 215 с.
- 12 Сырых Ю.А. Современный веб-дизайн. Рисуем сайт, который продает: учеб. пособие / Ю.А. Сырых. - М.: Вильямс, 2009. - 304 с.
- 13 Тихонов А.И. Публикация данных в Internet: учебник / А.И. Тихонов. - М.: Изд-во МЭИ, 2011. - 96 с.
- 14 Трофимов В.В. Информационные системы и технологии в экономике и управлении: учебник / В.В. Трофимов. - М.: Высшее образование, 2010. - 480 с.
- 15 Угринович Н.Д. Практикум по информатике и информационным технологиям / Н.Д. Угринович. - М.: Бином. Лаборатория занятий, 2012. - 400 с.
- 16 Программная среда для веб-разработки. OpenServer.: [Электронный ресурс], 2022 года – Режим доступа: <https://ospanel.io/> – 05.06.2022.
- 17 Денвер - локальный сервер. DENVER.: [Электронный ресурс], 2022 года – Режим доступа: <http://www.denwer.ru/> – 01.06.2022.
- 18 Когда display:none, а когда visibility: hiddenhtmlbook.ru.: [Электронный ресурс], 2022 года – Режим доступа: <http://htmlbook.ru/> – 25.04.2022.
- 19 HTML и Web дизайн для начинающих: [Электронный ресурс], 2022 года – Режим доступа: <http://sovet.h1.ru> – 29.04.2022.
- 20 Создание шаблона сайта с помощью CSS: [Электронный ресурс], 2022 года – Режим доступа: <http://ruseller.com> – 21.04.2022.

21 Создание страниц сайта с помощью HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://roumik.ru> – 21.04.2022.

22 Магия творчества. История вышивания: [Электронный ресурс], 2022 года – Режим доступа: <http://myhobby.net> – 21.04.2022.

23 Архитектурная мастерская. Создание сайта: [Электронный ресурс], 2022 года – Режим доступа: <http://wonderfulstyle.ucoz.com> – 21.04.2022.

24 Создание сайтов: HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://www.codeharmony.ru> – 21.04.2022.

25 Основы CSS и HTML: [Электронный ресурс], 2022 года – Режим доступа: <http://www.web-lesson.ru> – 21.04.2022.

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 17 |

Приложение А

Информационная модель

(обязательное)

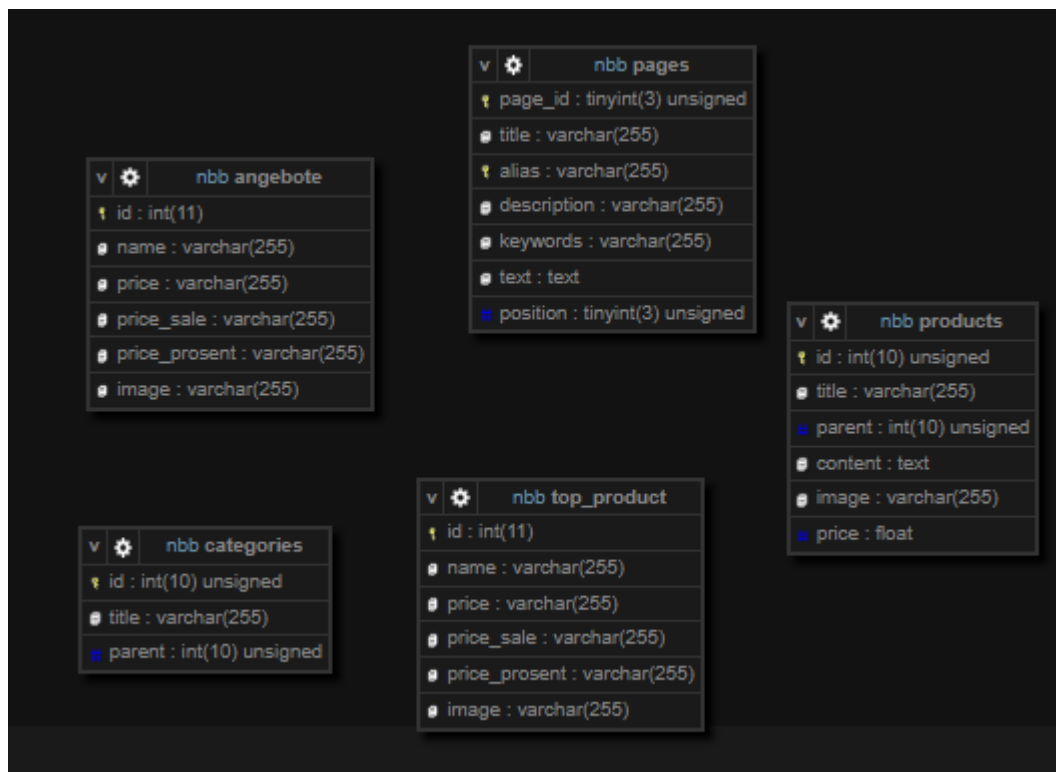


Рисунок А-1 – ER-диаграмма

Приложение Б

Функциональная модель (обязательное)



Рисунок Б-1 – Диаграмма контекстная диаграмма IDEF0

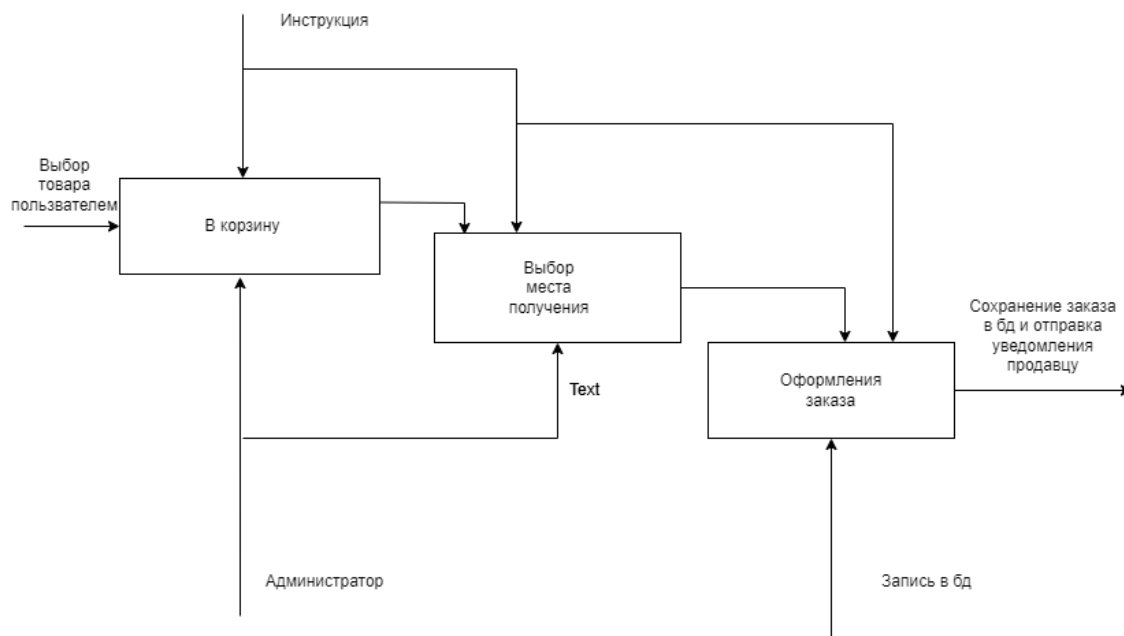


Рисунок Б-2 – Диаграмма декомпозиции первого уровня

Приложение В

Диаграмма прецедентов (обязательное)

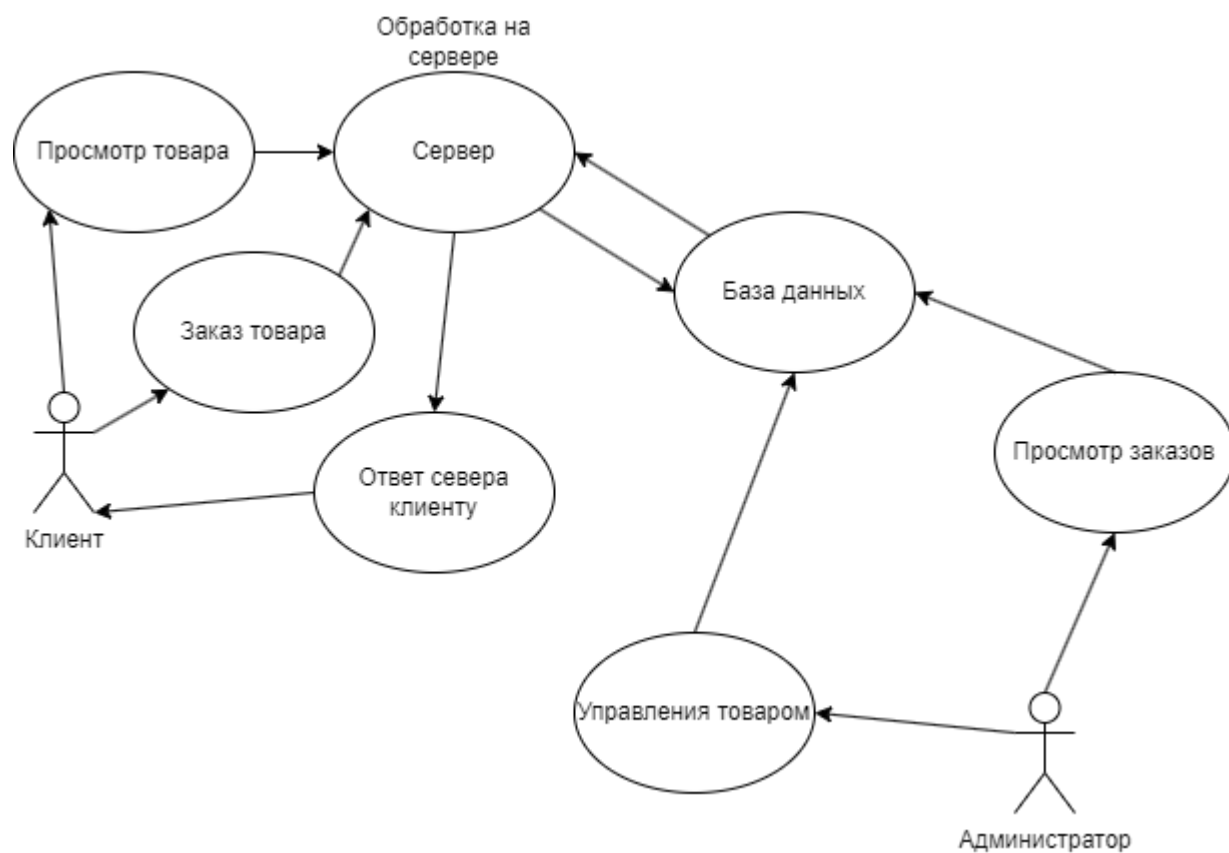


Рисунок Б-3 – Диаграмма претендентов интернет магазина nbb

Приложение Г Дизайн сайта (обязательное)

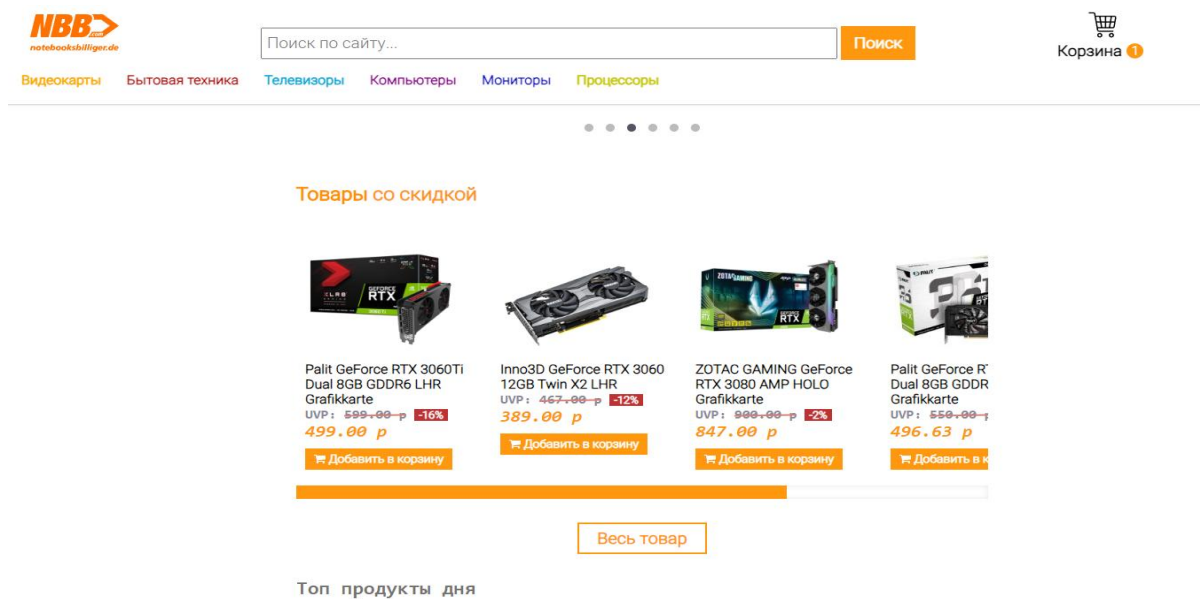


Рисунок Г-1 – Шаблон страницы главной страницы

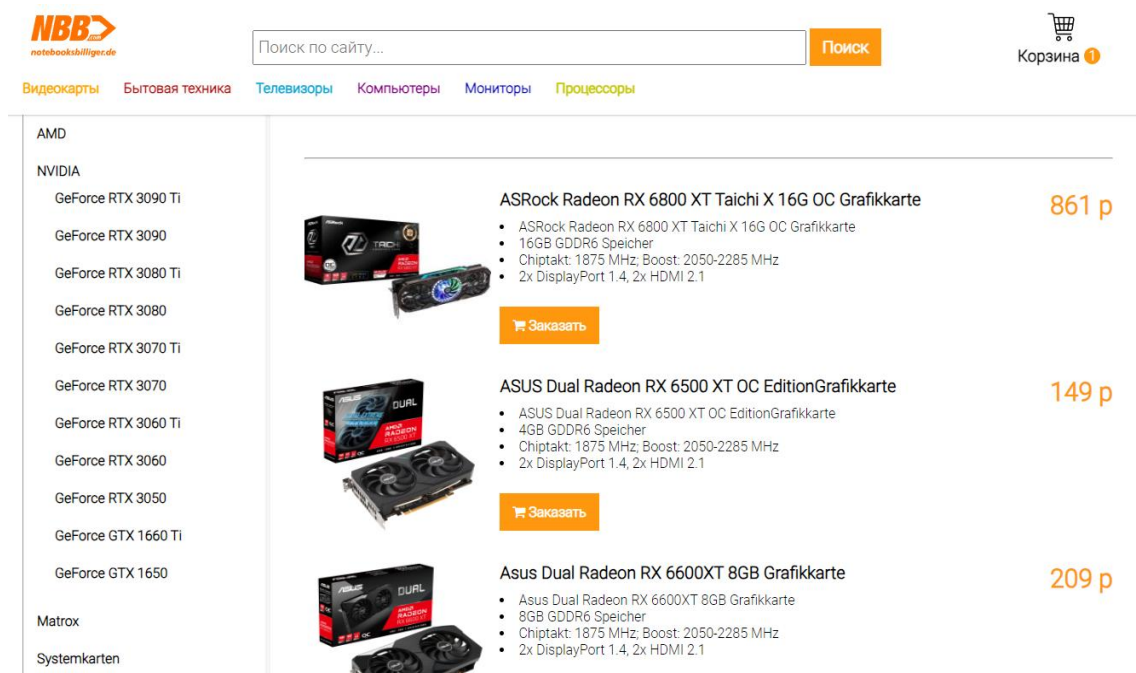


Рисунок Г-2 – Шаблон каталога

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 21 |

Рисунок Г-3 – Шаблон административной панели

Приложение Д

Листинг программы

(обязательное)

```
<?php

function print_arr($array){
    echo "<pre>" . print_r($array, true) . "</pre>";
}

function get_cat(){
    global $connection;
    $query = "SELECT * FROM categories";
    $res = mysqli_query($connection, $query);

    $arr_cat = array();
    while($row = mysqli_fetch_assoc($res)){
        $arr_cat[$row['id']] = $row;
    }
    return $arr_cat;
}

function map_tree($dataset) {
    $tree = array();

    foreach ($dataset as $id=> &$node) {
        if (!$node['parent']){
            $tree[$id] = &$node;
        }else{
            $dataset[$node['parent']]['childs'][$id] = &$node;
        }
    }

    return $tree;
}

function categories_to_string($data){
    foreach($data as $item){
        $string .= categories_to_template($item);
    }
    return $string;
}

function categories_to_template($category){
    ob_start();
    include 'category_template.php';
    return ob_get_clean();
}
```

```

function breadcrumbs($array, $id){
    if(!$id) return false;

    $count = count($array);
    $breadcrumbs_array = array();
    for($i = 0; $i < $count; $i++){
        if($array[$id]){
            $breadcrumbs_array[$array[$id]['id']] = $array[$id]['title'];
            $id = $array[$id]['parent'];
        }else break;
    }
    return array_reverse($breadcrumbs_array, true);
}

function cats_id($array, $id){
    if(!$id) return false;

    foreach($array as $item){
        if($item['parent'] == $id){
            $data .= $item['id'] . ",";
            $data .= cats_id($array, $item['id']);
        }
    }
    return $data;
}

function get_products($ids = false){
    global $connection;
    if($ids){
        $query = "SELECT * FROM products WHERE parent IN($ids) ORDER BY title";
    }else{
        $query = "SELECT * FROM products ORDER BY title";
    }
    $res = mysqli_query($connection, $query);
    $products = array();
    while($row = mysqli_fetch_assoc($res)){
        $products[] = $row;
    }
    return $products;
} ?>

```

```

<?php
include 'config.php';
include 'functions.php';
$categories = get_cat();
$categories_tree = map_tree($categories);
$categories_menu = categories_to_string($categories_tree);

if(isset($_GET['category'])){
    $id = (int)$_GET['category'];
    $breadcrumbs_array = breadcrumbs($categories, $id);
}

```



```

if($breadcrumbs_array){
    $breadcrumbs = "<a href='/catalog/'>Heimat</a> / ";
    foreach($breadcrumbs_array as $id => $title){
        $breadcrumbs .= "<a href='?category={$id}'>{$title}</a> / ";
    }
    $breadcrumbs = rtrim($breadcrumbs, " / ");
    $breadcrumbs = preg_replace("#(.+)?<a.+>(.)</a>#", "$1$2", $breadcrumbs);
}else{
    $breadcrumbs = "<a href='/catalog/'>Heimat</a> / catalog";
}

$ids = cats_id($categories, $id);
$ids = !$ids ? $id : rtrim($ids, ",");

if($ids) $products = get_products($ids);
    else $products = null;
}else{
    $products = get_products();
} ?>

```

```

function Sim(sldrId) {

    let id = document.getElementById(sldrId);
    if(id) {
        this.sldrRoot = id
    }
    else {
        this.sldrRoot = document.querySelector('.sim-slider')
    };

    // Slider objects
    this.sldrList = this.sldrRoot.querySelector('.sim-slider-list');
    this.sldrElements = this.sldrList.querySelectorAll('.sim-slider-element');
    this.sldrElemFirst = this.sldrList.querySelector('.sim-slider-element');
    this.leftArrow = this.sldrRoot.querySelector('div.sim-slider-arrow-left');
    this.rightArrow = this.sldrRoot.querySelector('div.sim-slider-arrow-right');
    this.indicatorDots = this.sldrRoot.querySelector('div.sim-slider-dots');

    // Initialization
    this.options = Sim.defaults;
    Sim.initialize(this)
};

Sim.defaults = {

    // Default options for the slider
    loop: true,    // Бесконечное заикливание слайдера
    auto: true,    // Автоматическое пролистывание
    interval: 5000, // Интервал между пролистыванием элементов (мс)
    arrows: true,  // Пролистывание стрелками
    dots: true     // Индикаторные точки

```

```

};

Sim.prototype.elemPrev = function(num) {
    num = num || 1;

    let prevElement = this.currentElement;
    this.currentElement -= num;
    if(this.currentElement < 0) this.currentElement = this.elemCount-1;

    if(!this.options.loop) {
        if(this.currentElement == 0) {
            this.leftArrow.style.display = 'none'
        };
        this.rightArrow.style.display = 'block'
    };

    this.sldrElements[this.currentElement].style.opacity = '1';
    this.sldrElements[prevElement].style.opacity = '0';

    if(this.options.dots) {
        this.dotOn(prevElement); this.dotOff(this.currentElement)
    }
};

```

```

Sim.prototype.elemNext = function(num) {
    num = num || 1;

    let prevElement = this.currentElement;
    this.currentElement += num;
    if(this.currentElement >= this.elemCount) this.currentElement = 0;

    if(!this.options.loop) {
        if(this.currentElement == this.elemCount-1) {
            this.rightArrow.style.display = 'none'
        };
        this.leftArrow.style.display = 'block'
    };

    this.sldrElements[this.currentElement].style.opacity = '1';
    this.sldrElements[prevElement].style.opacity = '0';

    if(this.options.dots) {
        this.dotOn(prevElement); this.dotOff(this.currentElement)
    }
};

```

```

Sim.prototype.dotOn = function(num) {
    this.indicatorDotsAll[num].style.cssText =
        'background-color:#BBB; cursor:pointer;'
};

```

```

Sim.prototype.dotOff = function(num) {
    this.indicatorDotsAll[num].style.cssText =

```

```

        'background-color:#556; cursor:default;';
};

Sim.initialize = function(that) {

    // Constants
    that.elemCount = that.sldrElements.length; // Количество элементов

    // Variables
    that.currentElement = 0;
    let bgTime = getTime();

    // Functions
    function getTime() {
        return new Date().getTime();
    };
    function setAutoScroll() {
        that.autoScroll = setInterval(function() {
            let fnTime = getTime();
            if(fnTime - bgTime + 10 > that.options.interval) {
                bgTime = fnTime; that.elemNext()
            }
        }, that.options.interval)
    };

    // Start initialization
    if(that.elemCount <= 1) { // Отключить навигацию
        that.options.auto = false;
        that.options.arrows = false; that.options.dots = false;
        that.leftArrow.style.display = 'none';
        that.rightArrow.style.display = 'none'
    };
    if(that.elemCount >= 1) { // показать первый элемент
        that.sldrElemFirst.style.opacity = '1';
    };

    if(!that.options.loop) {
        that.leftArrow.style.display = 'none'; // отключить левую стрелку
        that.options.auto = false; // отключить автопрокрутку
    }
    else if(that.options.auto) { // инициализация автопрокрутки
        setAutoScroll();
        // Остановка прокрутки при наведении мыши на элемент
        that.sldrList.addEventListener('mouseenter', function() {
            clearInterval(that.autoScroll)
        }, false);
        that.sldrList.addEventListener('mouseleave', setAutoScroll, false)
    };

    if(that.options.arrows) { // инициализация стрелок
        that.leftArrow.addEventListener('click', function() {
            let fnTime = getTime();
            if(fnTime - bgTime > 1000) {

```

```

        bgTime = fnTime; that.elemPrev()
    }
    }, false);
    that.rightArrow.addEventListener('click', function() {
        let fnTime = getTime();
        if(fnTime - bgTime > 1000) {
            bgTime = fnTime; that.elemNext()
        }
    }, false)
}
else {
    that.leftArrow.style.display = 'none';
    that.rightArrow.style.display = 'none'
};

if(that.options.dots) { // инициализация индикаторных точек
    let sum = "", diffNum;
    for(let i=0; i<that.elemCount; i++) {
        sum += '<span class="sim-dot"></span>'
    };
    that.indicatorDots.innerHTML = sum;
    that.indicatorDotsAll =
    that.sldrRoot.querySelectorAll('span.sim-dot');
    // Назначаем точкам обработчик события 'click'
    for(let n=0; n<that.elemCount; n++) {
        that.indicatorDotsAll[n].addEventListener('click', function(){
            diffNum = Math.abs(n - that.currentElement);
            if(n < that.currentElement) {
                bgTime = getTime(); that.elemPrev(diffNum)
            }
            else if(n > that.currentElement) {
                bgTime = getTime(); that.elemNext(diffNum)
            }
            // Если n == that.currentElement ничего не делаем
        }, false)
    };
    that.dotOff(0); // точка[0] выключена, остальные включены
    for(let i=1; i<that.elemCount; i++) {
        that.dotOn(i)
    }
}
};

```

```

import os
import time
class color:
    green = '\033[92m'
    red = '\033[91m'
    white = '\033[0m'
time.sleep(1)
os.system("clear")

```

[illegible]

```

os.system("clear")
time.sleep(1)
print("good bye")
exit(1)
else:
    time.sleep(1)
    os.system("clear")
    time.sleep(1)
    print(color.red + "Error DDosAttack!" + color.white)
    time.sleep(1)
    try1 = str(input("press Enter... "))
    if(str(choose) == "):
        os.system("python3 ddosattack.py")

```

| | | | | | | |
|------|------|----------|---------|------|--------------------------|------|
| | | | | | ОКЭИ 09.02.07. 9023 26 П | Лист |
| Изм. | Лист | № докум. | Подпись | Дата | | 30 |

