Государственное бюджетное профессиональное образовательное учреждение

Ростовской области

|  |
| --- |
| «Таганрогский колледж морского приборостроения» |

|  |
| --- |
| «Супер резинка» |
|  |
| Курсовой проект |
| Пояснительная записка |
| ТКМП.МДК.02.01.10.08 ПЗ |

|  |  |
| --- | --- |
| Руководитель  *Малыхина О.В.*  Студент П-320  *Ластовецкий А.М.* |  |
|  |

2023

**Содержание**

[Задание 3](#_Toc138779628)

[Аннотация 4](#_Toc138779629)

[Введение 5](#_Toc138779630)

[1. Общая часть 6](#_Toc138779631)

[1.1. Постановка задачи 6](#_Toc138779632)

[1.2. Анализ и исследование задачи, построение модели системы 8](#_Toc138779633)

[1.3. Разработка архитектуры решения 10](#_Toc138779634)

[1.4 Обоснование и выбор средств разработки решения 12](#_Toc138779635)

[1.5. Формализация расчетов 13](#_Toc138779636)

[1.5.1. Проектирование базы данных 15](#_Toc138779637)

[1.5.2. Проектирование интерфейса 17](#_Toc138779638)

[2. Описание программы 19](#_Toc138779639)

[3. Тестирование программы 20](#_Toc138779640)

[4 Программирование 22](#_Toc138779641)

[5 Отладка, оптимизация и тестирование программы 24](#_Toc138779642)

[Заключение 28](#_Toc138779643)

[Список используемых источников 29](#_Toc138779644)

[Приложение А. Результаты выполнения программы 30](#_Toc138779645)

**Задание**

Вашей задачей является разработка системы для производственной компании, которая ежедневно выпускает продукцию в большом объеме, а также реализует ее агентам, которые в дальнейшем продают эти товары конечным потребителям.

Для понимания объема разработки и планирования архитектуры приложения заказчик разделил весь проект на не сколько подсистем:

* подсистема для хранения информации о продукции,
* подсистема для работы со складом и материалами,
* подсистема для производства,
* подсистема по работе с сотрудниками,
* подсистема по работе с агентами.

Компания “Супер резинка” занимается производством банковских резинок, а также схожей эластичной продукции. Для нашей продукции мы используем множество вариантов сырья с различными свойствами для производства конечной продукции. Для понимания текущего состояния нашего склада материалов и его своевременного пополнения необходимо создать небольшую информационную систему.

Вам предстоит разработать подсистему для работы с материалами нашей компании, которая должна включать в себя следующий функционал:

* просмотр списка материалов,
* добавление/удаление/редактирование данных о материалах,
* управление списком возможных поставщиков материалов.

**Аннотация**

Данный курсовой проект представляет собой разработку веб-ресурса для управления системой складского хранения под названием "Супер резинка". Система складского хранения является важным компонентом эффективной логистики и управления запасами, поэтому разработка веб-ресурса позволит оптимизировать процессы учета, хранения и отслеживания товаров на складе. Основная цель проекта заключается в создании удобного и интуитивно понятного интерфейса, который позволит пользователям управлять системой складского хранения. Пользователи смогут осуществлять такие операции, как добавление товаров на склад, управление инвентаризацией, отслеживание статуса товаров, просмотр отчетов о запасах и перемещениях товаров на складе. Для достижения поставленной цели проекта будут использованы следующие технологии и инструменты: веб-разработка с использованием HTML, CSS и JavaScript для создания пользовательского интерфейса, серверная разработка с использованием языка программирования PHP для обработки бизнес-логики, а также база данных для хранения информации о продуктах, складских ячейках и операциях на складе.

# **Введение**

В современном мире, где электронная коммерция и онлайн-покупки становятся все более популярными, эффективная система складского хранения и управления запасами становится неотъемлемой частью успешного бизнеса. В контексте этого курсового проекта мы представляем вам веб-ресурс "Супер резинка" - инновационную систему складского хранения, разработанную с использованием языка PHP.

Цель данного курсового проекта заключается в создании веб-ресурса, который предоставит компаниям возможность эффективно управлять своими складскими операциями и оптимизировать процессы хранения и отгрузки товаров. Наша система предлагает набор инструментов и функциональных возможностей, которые помогут автоматизировать складские процессы, улучшить точность учета запасов и повысить общую эффективность бизнеса.

Основная особенность "Супер резинки" заключается в ее разработке на языке программирования PHP. PHP - это широко используемый язык программирования, особенно в веб-разработке, благодаря своей гибкости, простоте и мощности. Использование PHP позволяет нам создавать динамические веб-страницы, взаимодействовать с базами данных и разрабатывать функциональные модули для обработки заказов, управления запасами и многого другого.

**1. Общая часть**

**1.1. Постановка задачи**

Постановка задачи:

Целью данного курсового проекта является разработка веб-ресурса под названием "Супер резинка" - системы складского хранения и управления запасами, используя язык программирования PHP. Веб-ресурс "Супер резинка" предназначен для помощи компаниям в эффективном управлении и контроле их складских операций.

Для достижения данной цели необходимо решить следующие задачи:

1. Разработка пользовательского интерфейса:

Создать интуитивно понятный и удобный пользовательский интерфейс, который позволит пользователям легко взаимодействовать с системой. Интерфейс должен быть отзывчивым и иметь интуитивно понятные элементы управления для выполнения различных задач, таких как учет товаров, отслеживание запасов и генерация отчетов.

1. Учет и отслеживание товаров:

Реализовать функционал, который позволит пользователям добавлять новые товары, удалять или изменять существующие, а также отслеживать их количество и доступность на складе. Должна быть возможность просматривать информацию о товаре, такую как описание, цена, поставщик и прочее.

1. Управление запасами:

Создать механизм, позволяющий автоматически оповещать пользователя о низких уровнях запасов и предлагать возможности для пополнения запасов. Система должна предлагать оптимальные рекомендации по дозаказу товаров, основываясь на уровне спроса и заданных параметрах запасов.

1. Оптимизация распределения товаров:

Реализовать функционал, который поможет оптимизировать распределение товаров по различным зонам склада, учитывая их характеристики, требования к хранению и скорость продажи. Это позволит эффективно использовать пространство склада и снизить затраты на хранение.

**1.2. Анализ и исследование задачи, построение модели системы**

Анализ задачи:

Для начала необходимо проанализировать основные требования к системе складского хранения "Супер резинка". Основной целью данного веб-ресурса должно быть обеспечение удобства и эффективности работы с продукцией на складе.

Основные задачи, которые должны быть решены с помощью системы складского хранения, включают:

Учет и контроль за поступлением и отгрузкой товаров;

Мониторинг остатков и движения товаров на складе;

Поддержание актуальности информации о товарах и их расположении на складе;

Управление заказами на доставку товаров.

Исследование технологий:

Для создания веб-ресурса системы складского хранения можно использовать различные языки программирования и фреймворки. Наиболее распространенным языком программирования для разработки веб-приложений является PHP. Этот язык предоставляет широкие возможности для создания динамических веб-сайтов с использованием баз данных MySQL.

Построение модели системы:

Для построения модели системы необходимо определить ее структуру и функциональные возможности. Разработка системы складского хранения должна включать следующие компоненты:

База данных для управления информацией о товарах, клиентах и остатках;

Интерфейс пользователя для взаимодействия с системой;

Механизмы обработки заказов на доставку товаров;

Механизмы мониторинга остатков и движения товаров на складе.

База данных должна содержать информацию о каждом товаре, включая его наименование, описание, цену, количество и расположение на складе. Кроме того, база данных должна содержать информацию о клиентах, заказах на доставку товаров и остатках на складе.

Интерфейс пользователя должен предоставлять возможность просмотра информации о товарах, оформления заказов на доставку, а также мониторинга остатков товаров на складе.

Механизмы обработки заказов на доставку должны включать автоматическое формирование накладных и документов для отслеживания поставок товаров.

Механизмы мониторинга остатков и движения товаров на складе должны обеспечивать точный и актуальный учет движения товаров на складе, а также предоставлять отчеты для анализа эффективности работы системы.

Таким образом, создание веб-ресурса системы складского хранения "Супер резинка" с использованием языка программирования PHP позволит обеспечить эффективное управление продукцией на складе, повысить его эффективность и сохранить актуальную информацию о товарах.

## **1.3. Разработка архитектуры решения**

Для разработки подсистемы для работы с товарами веб-ресурса “Супер резинка” необходимо определить ее архитектуру. Архитектура системы должна обеспечивать исключительную производительность, надежность и безопасность при работе с данными клиентов и товаров сайта. В качестве архитектурного решения можно применить следующие подходы:

1. Микросервисная архитектура:

Этот подход предполагает разбиение приложения на независимые микросервисы, каждый из которых отвечает за выполнение отдельной функциональности. Для работы с товарами можно выделить отдельный микросервис, который будет обеспечивать определенный функционал работы с товарами. Каждый микросервис может иметь свой собственный индивидуальный набор взаимодействующих с ним процессов и элементов, что снижает вероятность возникновения сбоев и повышает отказоустойчивость системы.

1. Клиент-серверная архитектура:

Это классический подход в веб-разработке, который предполагает разделение приложения на клиентскую и серверную части. Основная логика и данные хранятся на сервере, а клиентская часть отвечает за отображение информации и взаимодействие с пользователем. В данной архитектуре в качестве серверной части может выступать отдельный сервер, на котором будет развернуто приложение для работы с товарами. Клиентская часть приложения будет предоставлять пользователю возможность просматривать, добавлять, удалять и редактировать данные о товарах.

1. Эволюционная архитектура:

Она представляет собой подход к разработке ПО, основанный на постепенном приращении функционала системы. Компоненты и модули системы постепенно добавляются к уже имеющейся инфраструктуре. Для разработки подсистемы для работы с товарами можно начать с небольшого приложения, которое будет обеспечивать просмотр списка товаров и истории покупок. Дополнительный функционал, такой как добавление/удаление/редактирование данных о товарах и предложения необходимых запчастей к основному товару, может быть добавлен в последующих этапах разработки. При выборе одного из этих подходов нужно учитывать требования к производительности, ограничения базы данных, требования к защите данных и возможности разработчиков. Комбинация различных архитектурных подходов может обеспечить наилучший баланс между функциональностью, производительностью и безопасностью системы.

## **1.4 Обоснование и выбор средств разработки решения**

Выбранное средство разработки решения для данного курсового проекта на тему "Супер резинка" - веб-ресурс для учета товаров нашего сайта "Супер резинка" - состоит из трех основных компонентов: HTML, CSS и JavaScript. Наиболее подходящим и распространенным инструментом, который обеспечивает все эти возможности является PHP. Выбор средства разработки веб-ресурса для данного проекта сделан, исходя из следующих причин:

1. HTML - это стандартный язык разметки веб-страниц, который является основным языком для создания всей структуры и функциональности веб-сайта.

2. CSS - это язык таблиц стилей, который позволяет определить стиль веб-страницы, изменять ее внешний вид и улучшать пользовательский опыт. 3. JavaScript - это язык программирования, который позволяет создавать интерактивные элементы веб-страниц, улучшать навигацию по сайту и предоставлять пользователю средства для взаимодействия со страницей. 4. PHP - это скриптовый язык программирования, который является очень распространенным веб-средством для создания и управления базами данных.

5. База данных сделана через XAMPP. Это кроссплатформенный дистрибутив, который может использоваться на Unix, Linux, Mac и Windows. XAMPP включает в себя Apache, PHP, MySQL и Perl, что делает его идеальным выбором для разработки веб-приложений.

6. PHP является наиболее распространенным языком программирования для создания веб-приложений с открытым исходным кодом. Он поддерживает работу с большинством баз данных, включая MySQL, SQLite, PostgreSQL и другие. В результате, выбор использования CSS, PHP и JavaScript оправдан, так как они позволяют создавать качественный и функциональный веб-ресурс для работы с товарами на сайте. Кроме того, использование XAMPP позволит обеспечить надежность и стабильность базы данных и приложения в целом.

## **1.5. Формализация расчетов**

Для формализации расчетов для разработки подсистемы учета товаров на веб-ресурсе "Супер резинка" необходимо определить следующие шаги:

1. Определить ожидаемые результаты работы подсистемы: - Возможность просмотра списка товаров - Возможность добавления/удаления данных о товарах - Возможность предложения дополнительных товаров к основным - Возможность просмотра истории покупок товаров клиента.
2. Определить методы передачи данных: - Использование базы данных через xampp - Использование языков программирования: css, php, javascript.
3. Определить логику работы подсистемы: - Агенты могут просматривать список товаров и делать заказы - Администраторы предприятия могут добавлять/удалять данные о товарах и предложения к ним - Подсистема должна иметь возможность формирования отчетов о покупках товаров предприятия.
4. Определить использование инструментов программирования: - Для создания интерфейса можно использовать языки программирования HTML и CSS - Для обработки запросов и отображения данных необходимо использовать сценарии на языке программирования PHP - Для реализации динамической интерактивности можно использовать языки программирования AJAX и JavaScript.
5. Определить требования к базе данных: - Необходимо определить таблицы хранения информации о товарах и предложениях к ним - Необходимо определить таблицы хранения информации о покупках предприятия.
6. Определить и описать процессы и взаимодействия между компонентами системы: - Агенты должны иметь возможность через веб-интерфейс просматривать список товаров и оформлять покупку. После этого данные о заказе должны попасть в базу данных. - Администраторы должны иметь возможность через интерфейс управления добавлять/удалять данные о товарах и описания к ним. - Подсистема должна иметь возможность формирования списка покупок предприятия и вывода этой информации на экран. С учетом этих шагов необходимо разработать программный код и написать документацию по проекту.

### **1.5.1. Проектирование базы данных**

Для проектирования базы данных для веб-ресурса «Супер резинка» необходимо следовать определенной структуре. При этом можно использовать реляционную модель. Следующие таблицы могут быть использованы в данной базе данных:

1. Таблица "Пользователи"

- ID пользователя (уникальный идентификатор)

- ID группы пользователя (уникальный идентификатор группы)

- Имя пользователя

- Пароль пользователя

- Адрес электронной почты

- Блокировка пользователя

2. Таблица "Продукты"

- ID продукта (уникальный идентификатор)

- Название товара

- Материалы товара

- Количество товара на складе

- Изображение продукта

3. Таблица "Материалы"

- ID материала (уникальный идентификатор)

- Название материала

4. Таблица "Корзина"

- ID (уникальный идентификатор)

- ID продукта (уникальный идентификатор)

- ID агента (уникальный идентификатор)

- Количество заказанного товара

Для связи этих таблиц можно использовать сводные ключи:

- ID агента из таблицы "Пользователя" связывается с ID покупки из "Корзины"

- ID товара из таблицы "Продукты" связывается с ID покупки из "Корзины

С использованием PHP и JavaScript можно разработать интерфейс для взаимодействия с базой данных для выполнения вышеперечисленных функций. Просмотр списка товаров может происходить путем запроса на получение всех записей из таблицы "Продукты" и отображения их на странице. Добавление, удаление и изменение данных о товарах может осуществляться при помощи SQL-запросов на добавление, удаление и обновление данных в таблице "Товары". Предложения дополнительных товаров могут быть сделаны путем добавления новых записей в таблицу "Продукты" с признаком связи с другим товаром. Просмотр истории покупок может быть осуществлен путем запроса на получение всех записей из таблицы "Корзина" и отображения их на странице. Для удобства учета товаров администраторы сервиса могут использовать специальную админ-панель, где будут представлены все вышеперечисленные функции. Таким образом, разработка подсистемы для работы с товарами имеет несколько ключевых моментов: создание таблицы товаров, добавление, удаление, редактирование данных о товарах, предложения дополнительных товаров к основным и просмотр истории покупок товаров автосервиса. База данных может быть создана с помощью XAMPP, а интерфейс взаимодействия с базой данных может быть разработан с использованием PHP и JavaScript.

### **1.5.2. Проектирование интерфейса**

Для успешного проектирования интерфейса подсистемы по работе с товарами необходимо учитывать следующие пункты:

1. Навигация по сайту должна быть простой и удобной. Для этого можно создать панель навигации, которая будет содержать ссылки на все подсистемы.

2. Главная страница должна содержать основную информацию о компании и ее услугах. Также на главной странице можно разместить список наиболее популярных товаров.

3. Страница со списком товаров должна содержать таблицу, которая будет включать следующие поля: название товара, стоимость, наличие, возможность заказа и редактирования.

4. Для добавления нового товара можно создать форму, которая будет содержать необходимые поля для заполнения: название, описание, стоимость, количество.

5. Для удаления товара из базы данных можно создать кнопку "Удалить". При нажатии на данную кнопку товар должен удалиться из базы данных. 6. Для редактирования данных о товаре можно создать форму, которая будет содержать все поля, заполненные текущими данными. Для изменения необходимых данных необходимо изменить соответствующие поля и нажать кнопку "Сохранить".

7. Предложения дополнительных товаров к основным могут быть оформлены в виде блока, который будет содержать подсказки с ссылками на дополнительные товары.

8. История покупок товаров автосервиса может быть оформлена в виде таблицы. Таблица должна содержать следующие поля: название товара, количество, дата покупки, общая сумма покупки.

9. Весь интерфейс должен быть адаптивным и подходить для использования на различных устройствах с разными разрешениями экранов.

10. Необходимо также придерживаться общих стандартов и требований к дизайну и интерфейсу веб-страниц, включая использование читабельных шрифтов, цветовую гамму, логотип, и т.д. Написание веб-ресурса на основе CSS, PHP, JavaScript и базы данных созданной в XAMPP, позволит создать профессиональный, удобный и функциональный интерфейс для подсистемы по работе с товарами нашей компании.

# **2. Описание программы**

Программа, разработанная для предприятия "Супер резинка", является веб-ресурсом, который оказывает услуги по распределению товаров предприятия между агентами. В соответствии с этой задачей, была разработана система, которая включает в себя подсистемы по работе с клиентами, сотрудниками, товарами, услугами и для управления предприятием.

Специальное внимание уделено разработке подсистемы для работы с товарами, которая состоит из следующего функционала:

- Просмотр списка товаров, где агенты могут ознакомиться с наличием товаров и их ценами.

- Добавление/удаление данных о товарах. Администраторы могут легко добавлять новые товары, редактировать информацию о существующих и удалять товары, которые больше не нужны.

- Предложения дополнительных товаров к основным. Пользователи могут ознакомиться с дополнительными предложениями, связанными с основными товарами, которые они уже выбрали. Это помогает агентам сделать более информированный выбор и улучшить их опыт работы с компанией "Супер резинка".

Для разработки веб-ресурса были использованы следующие технологии: CSS, PHP и JavaScript. База данных была создана с помощью XAMPP. Эти технологии помогли разработчикам создать простой в использовании и удобный веб-ресурс для компании, который позволяет клиентам и сотрудникам быстро получать доступ к нужным информационным и техническим подсистемам.

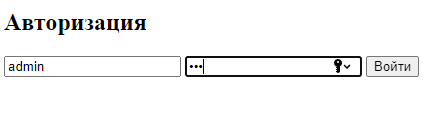
# **3. Тестирование программы**

Способ проведения тестирования:

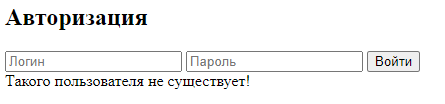
1. Проверить работоспособности авторизации
2. Проверить работоспособность функций товара
3. Проверить работоспособность функции выхода из аккаунта

Тестирование:

1. Для проверки авторизации мы введём логин и пароль в соответствующие поля и нажмём кнопку «Войти». Пример авторизации в соответствии с рисунком 5.1. Если ввести неправильный логин или пароль, то система выдаст ошибку в соответствии с рисунком 5.2.

****

**Рисунок 1.1 Авторизация**

****

**Рисунок 1.2 Ошибка авторизации**

2)Для проверки работоспособности функций заказа товара мы сначала должны выбрать вариант, который нам подходит.



**Рисунок 1.3 Оформление заказа**

# **4 Программирование**

Система автосервиса спроектирована на языке PHP и БД MySQL.

Описание запросов MySQL:

1. Создание таблицы «Корзина».

CREATE TABLE `card` (

`id` int(11) NOT NULL,

`id\_product` int(11) NOT NULL,

`id\_agent` int(11) NOT NULL,

`quantity` text NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_general\_ci;

Создание таблицы «Материалы».

CREATE TABLE `material` (

`id` int(11) NOT NULL,

`name` varchar(100) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_general\_ci;)

Создание таблицы «Продукты».

CREATE TABLE `product` (

`id` int(11) NOT NULL,

`name` varchar(100) NOT NULL,

`materials` text NOT NULL,

`quantity` text NOT NULL,

`product\_img` text NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_general\_ci;)

Создание таблицы «Пользователи».

CREATE TABLE `user` (

`id` int(11) NOT NULL,

`idgroup` int(11) NOT NULL,

`login` varchar(100) NOT NULL,

`password` text NOT NULL,

`email` text NOT NULL,

`blocked` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_general\_ci;

Сценарии в коде программы:

1. login.php – Авторизация

Входные данные: логин и пароль  
Выходные данные: вход на сайт

1. delete\_таблица.php – Удаление строки из таблицы

Входные данные: ID строки  
Выходные данные: удалённая строка

1. update\_таблица.php – Изменение строки в таблице

Входные данные: ID строки и новые данные  
Выходные данные: изменённая строка

1. db.php – Подключение к базе данных

Входные данные: Название БД, пароль, имя учётной записи phpMyAdmin  
Выходные данные: Данные в таблицах

# **5 Отладка, оптимизация и тестирование программы**

Способ проведения тестирования:

1. Проверить работоспособности авторизации
2. Проверить работоспособность функций записи и удаления продуктов
3. Проверить работоспособность функции добавление, блокировку и снятие блокировки с пользователей
4. Проверить работоспособность функции выхода из аккаунта

Тестирование:

1. Для проверки авторизации мы введём логин и пароль в соответствующие поля и нажмём кнопку «Войти». Пример авторизации в соответствии с рисунком 5.1. Если ввести неправильный логин или пароль, то система выдаст ошибку в соответствии с рисунком 5.2.

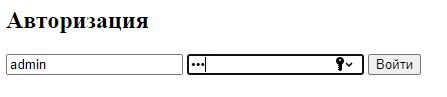


Рисунок 1.4 – Авторизация

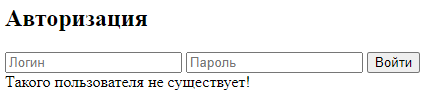


Рисунок 1.5 – Ошибка авторизации

1. Для проверки функций во вкладке «Заказать» введём данные в поля для добавления заказа и нажмём кнопку «Заказатб» в соответствии с рисунком 1.6.



Рисунок 1.6 – Оформление заказа



Рисунок 1.7 – Новый заказ

1. Для проверки возможности добавления товара во вкладке «Добавить товар» войдем под логином администратора и добавим товар нажав на значок добавления продукта.

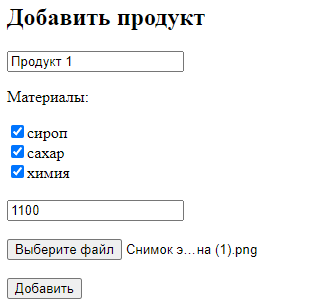


Рисунок 1.9 – Добавленный продукт

После нажатия кнопки, данные на странице изменились. Продемонстрировано на рисунке 5.7.

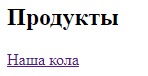


Рисунок 2.1 – Изменённая страница продуктов

Для проверки удаления данных со страницы нажмём на кнопку «Удалить» в соответствии с рисунком 2.2.



# **Заключение**

В процессе разработки и работы над курсовым проектом была поставлена задача – реализовать программу для ведения веб-сайта “Чистый дом”. Данная задача была проанализирована, спроектирована, построена и выполнена. Были изучены и представлены примеры похожих программ, также был описан и спроектирован принцип работы программы и произведено тестирование исходной программы. Данный проект имеет недостатки, но также может быть доработан и улучшен.

# **Список используемых источников**

1. <https://metanit.com/php/tutorial/>
2. <https://metanit.com/web/html5/>
3. <https://metanit.com/sql/>
4. <https://metanit.com/sql/mysql/>

# **Приложение А. Результаты выполнения программы**

При входе на сайт нас встречает окно авторизации, предназначенное для Покупателя сайта в соответствии с рисунком 2.3.

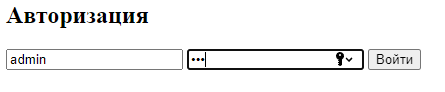


Рисунок 2.3 – Окно авторизации

После ввода логина и пароля нужно нажать кнопку «Войти», клиента переносит на страницу с товарами в соответствии с рисунком 2.4

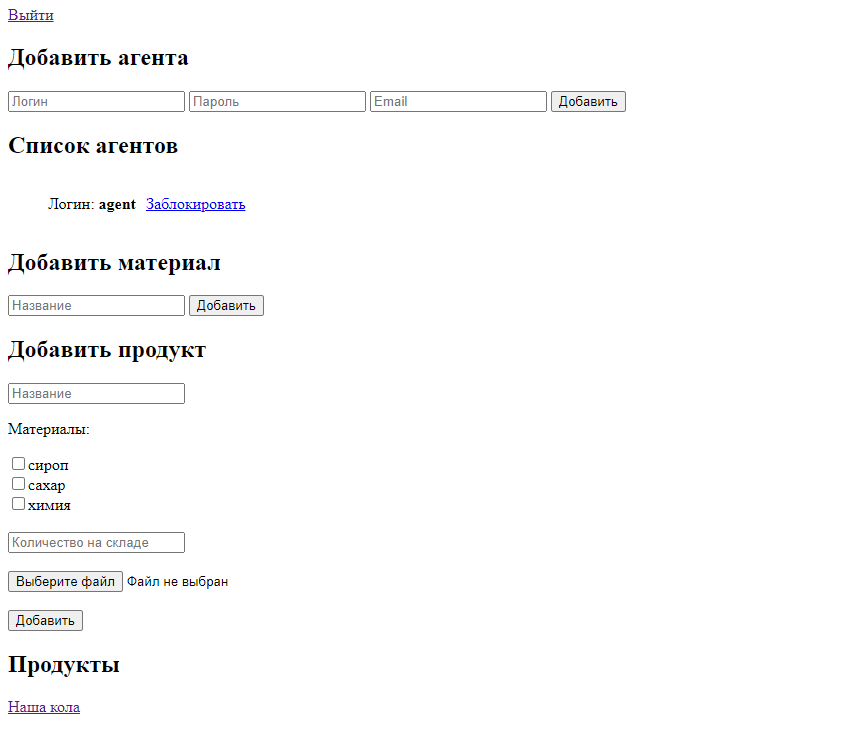


Рисунок 2.4- Страница с товарами

Далее есть возможность оформить заказ товара на вкладке «Заказать» в соответствии с рисунком 2.5



Рисунок 2.4- Страница оформления заказа