Поволжский Государственный Университет Телекоммуникаций и Информатики  
  
Кафедра «Программная инженерия»

|  |  |
| --- | --- |
| Сдана на проверку  «\_\_» \_\_\_\_2023 г. | Допустить к защите  «\_\_» \_\_\_\_\_ 2023 г.  Защищена с оценкой  «\_\_\_» \_\_\_\_\_\_\_\_ 2023 г. |

КУРСОВАЯ РАБОТА  
По дисциплине: «Прикладное программирование»  
На тему: «Разработка клиент-серверного web-приложения «Учётка» для учёта наличия товаров»  
  
Пояснительная записка

|  |  |
| --- | --- |
| Студент группы РПИС-11 \_\_\_\_\_\_\_ Gosling R.T.  (роспись) (ФИО)  Руководитель \_\_\_\_\_\_\_\_\_\_ к.т.н., доц. Ахметшина Э.Г.  (роспись) (ФИО) | 0 1488 0  (№ зачетной книжки) |

Самара 2023

# **Рецензия**

Добавить везде форматирование по ширине; убрать лишние межстрочные интервалы; добавить отступы абзаца 1.25; добавить межстрочный интервал 1.5;  
обновить диаграмму классов.

**Содержание**

1. [Описание предметной области](#Описание_предметной_области). Актуальность……………………………......4

2. [Описание программы](#Описание_программы). Общие сведения.………………………………..……..5

[3. Функциональное назначение 5](#_Toc154396745)

[4. Логическая модель базы данных 7](#_Toc154396746)

[5. Физическая модель базы данных 8](#_Toc154396747)

[6. Диаграмма классов 9](#_Toc154396748)

[7. Диаграмма компонентов 10](#_Toc154396749)

[8. Диаграмма вариантов использования 10](#_Toc154396750)

[9. Диаграмма последовательности 11](#_Toc154396751)

[10. Демонстрация работы приложения 12](#_Toc154396752)

[Список используемых источников 19](#_Toc154396753)

[Приложение – Листинг программного кода 20](#_Toc154396754)

# **1. Описание предметной области. Актуальность**

Современная деловая среда все более активно внедряет информационные технологии для улучшения процессов управления и контроля за инвентаризацией товаров. В этом контексте создание веб-приложения для учета товаров, сортировки по категориям и отслеживания операций продаж и покупок играет важную роль в оптимизации бизнес-процессов.

Рост интереса к цифровым инструментам учета и управления товарными запасами является следствием динамичного развития электронной коммерции и сферы предпринимательства в целом. В настоящее время компании и предприниматели стремятся эффективно управлять своими инвентарными запасами, улучшать обзорность и контроль за наличием товаров различных категорий.

Российский рынок электронной коммерции демонстрирует стабильный рост объемов онлайн-торговли. Согласно данным аналитических исследований, за прошлый год отмечен значительный прирост объемов онлайн-коммерции в секторе потребительских товаров на 16%. Общий объем рынка розничной интернет-торговли в России значительно увеличился, достигнув 5,7 трлн рублей, а количество заказов возросло на 65%.

Тенденция увеличения числа покупателей, предпочитающих совершать покупки через онлайн-платформы, является важным фактором для разработки и внедрения веб-приложения, способного удовлетворить потребности клиентов в удобном учете товаров, их категоризации и отслеживании операций купли-продажи.

Цель данной курсовой работы заключается в разработке веб-приложения «Учётка» для удобного ведения информации о наличии товаров, их классификации по категориям и возможности внесения записей о продажах и покупках. Для реализации данного проекта был использован фреймворк Spring, обладающий необходимым функционалом для создания клиент-серверного веб-приложения.

Интернет-приложение, представленное в данной работе, открывает перед предпринимателями и компаниями возможность эффективного управления запасами, оптимизации процессов продаж и улучшения работы с клиентами. Развитие подобных веб-ресурсов считается перспективным, поскольку они обеспечивают не только удобство в учете товаров, но и способствуют расширению возможностей торговли за счет доступности и удобства онлайн-платформ.

**2. Описание программы. Общие сведения**

Приложение – веб-ресурс. Установка не требуется.

Необходимые требования: Web-браузер (любой).

Язык программирования: Java.

Платформа: Spring Boot.

Среда разработки: IntelliJ IDEA.

Объем проекта: 1.8 МБ.

Исходный код: 1743‬ строки.

# **3. Функциональное назначение**

Функциональное назначение клиент-серверного web-приложения учёта наличия товаров и записей продаж/покупой «Учётка» включает следующие возможности:

* Просмотр всех товаров;
* Редактирование товаров;
* Удаление товаров
* просмотр товаров по категориям;
* просмотр категорий товаров;
* редактирование категорий;
* удаление категорий;
* запись продаж/покупок;
* просмотр истории продаж/покупок;

При разработке клиент-серверного web-приложения использовались следующие технологии: Spring Boot и thymeleaf для реализации взаимодействия между сервером и клиентом через протокол HTTP, Spring Boot использует REST API для формирования динамического контента на веб-страницах. В качестве локального сервера web-приложения использовался встроенные в Spring Boot - Tomcat. Взаимодействие сервера с БД MySQL было реализовано через JDBC. Для разработки клиентской части использовались технологии HTML, CSS, JavaScript.

Spring Boot - фреймворк для разработки веб-приложений, который предоставляет простой способ создания мощных приложений. Он основывается на платформе Spring и упрощает конфигурацию, позволяя разработчикам быстро создавать приложения с использованием Java. Spring Boot обеспечивает удобную интеграцию с различными инструментами, а также предоставляет решения для ряда задач, включая взаимодействие с базами данных, обработку запросов и многое другое.

Thymeleaf - это мощный шаблонизатор, интегрированный с Spring Framework, который позволяет создавать динамические веб-страницы с использованием HTML, CSS и JavaScript. Thymeleaf упрощает работу с данными и обеспечивает удобные инструменты для их отображения и обработки. Он обеспечивает гибкую интеграцию с Spring Boot, позволяя создавать адаптивные пользовательские интерфейсы и управлять данными в приложении.

Для обработки запросов и предоставления динамического контента на страницах, наше приложение использует связку Spring Boot и Thymeleaf. Spring Boot обрабатывает входящие запросы, взаимодействует с базой данных и управляет бизнес-логикой приложения, в то время как Thymeleaf отвечает за генерацию HTML-кода на основе шаблонов, которые содержат данные, передаваемые контроллерами. Такая комбинация позволяет нам создавать динамичные и отзывчивые веб-страницы для удобного учета наличия товаров.

Для создания статической составляющей веб-страниц применялся язык гипертекстовой разметки HTML, для стилизации страниц были использованы каскадные таблицы стилей CSS (Cascading Style Sheets). Для создания динамической составляющей страниц использовался язык JavaScript.

# **4. Логическая модель базы данных**

Предметная область: база данных обеспечивает операции добавления, удаления и изменения над товарами, их категориями и записями продаж/покупок. Каждая из этих функций доступна на веб странице и человек, ответственный за это, сможет быстро заполнить необходимые данные.

В этой БД имеются сущности:

1. Товары;
2. Категории;
3. Инвентаризация.

Атрибуты сущностей:   
1. Товары (Products) – хранит ID товара, его наименование, описание, айди категории, количество и изображение этого товара.   
2. Категории (Categories) – хранит ID категории, название этой категории, описание и изображение.  
3. Инвентаризация (Inventory\_Records) – хранит ID действия, ID продукта, количество, дату и название действия.

Связи атрибутов сущностей:

1. Категории (Categories) и Товары (Products):
   * Сущность "Категории" (Categories) связана с сущностью "Товары" (Products) через внешний ключ category\_id в таблице "Товары" и первичный ключ id в таблице "Категории".
   * Связь между "Категориями" и "Товарами" установлена как "Один ко многим", где каждая категория может иметь много товаров, но каждый товар принадлежит только одной категории.
2. Инвентаризация (Inventory Records) и Продукты (Products):
   * Сущность "Инвентаризация" (Inventory Records) связана с сущностью "Товары" (Products) через внешний ключ product\_id в таблице "Инвентаризация " и первичный ключ id в таблице "Товары".
   * Эта связь устанавливает отношение "Многие к одному", где множество записей инвентаризации могут ссылаться на один и тот же товар.
3. Товары (Products) и Категории (Categories):
   * Сущность "Товары" (Products) содержит внешний ключ category\_id, который связывает каждый продукт с конкретной категорией из сущности "Категории" (Categories).
   * Это отношение также представляет собой связь "Многие к одному", где каждый продукт принадлежит только одной категории, а у каждой категории может быть множество продуктов.

# **5. Физическая модель базы данных**

Физическая модель базы данных — это модель данных, которая определяет, каким образом представляются данные, и содержит все детали, необходимые СУБД для создания базы данных.

На рисунке 1 отображена физическая модель базы данных для связи products и inventory\_records.

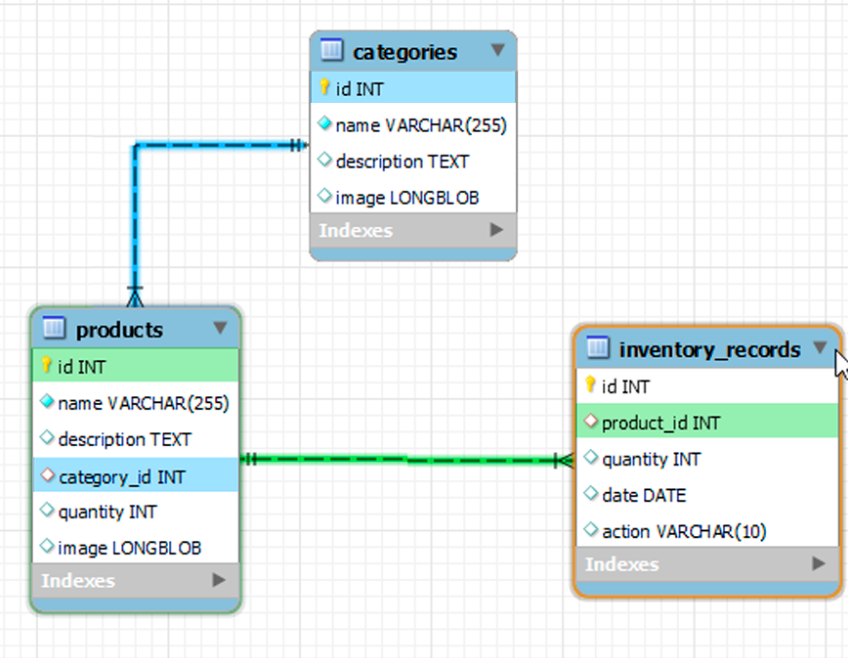


Рис. 1 – Физическая модель БД

# **6. Диаграмма классов**

Диаграмма классов в языке моделирования UML служит для отображения общей структуры и иерархии классов в системе, включая их атрибуты, методы, интерфейсы и взаимосвязи между ними. Этот вид диаграммы представляет статические и динамические аспекты системы, включая не только описание классов, атрибутов и методов, но и взаимосвязи между объектами и их поведение в процессе выполнения программы.

Такая диаграмма соответствует основным принципам объектно-ориентированного программирования (ООП) и является важным инструментом проектирования систем, описывая их структуру и взаимодействие между компонентами. Она помогает визуализировать организацию классов и их взаимосвязи, обеспечивая лучшее понимание архитектуры системы на уровне объектов и классов.

Диаграмма классов показана на рисунке 2.

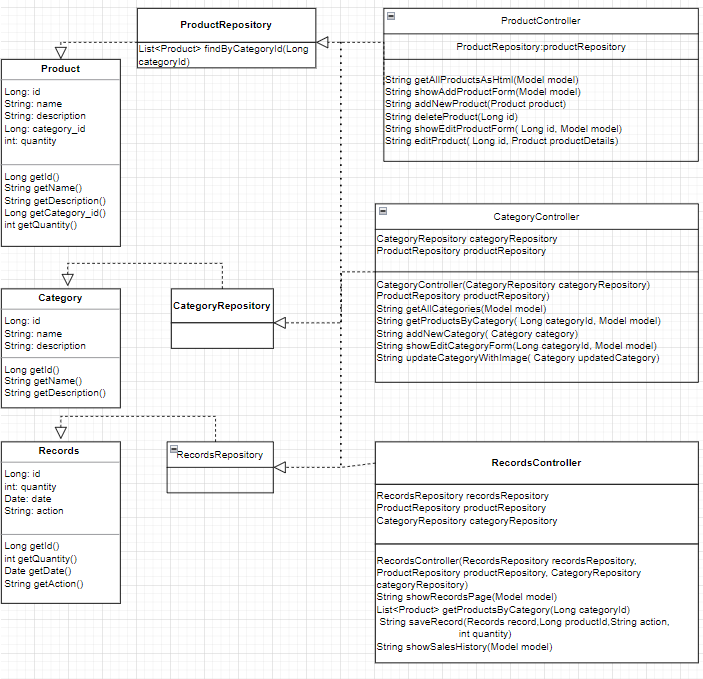


Рис. 2 – Диаграмма классов

С клиента поступает запрос на сервер, сервер обрабатывает запрос и обращается к классу «Controller». Класс «Controller» обращается к нужному ему entity классу, который обращается в нужный ему interface и отправляет запрос в базу данных. БД отправляет ответ, проходя по тому же маршруту, сервер получает ответ и отправляет клиенту.

# **7. Диаграмма компонентов**

Диаграмма компонентов клиент-серверного web-приложения для учёта наличия товаров «Учётка» показывает взаимодействие между следующими компонентами: клиентский веб-интерфейс, серверный компонент и база данных. Клиентский веб-интерфейс отвечает за отображение пользовательского интерфейса и взаимодействие с пользователями, серверный компонент отвечает за обработку запросов от клиента и взаимодействие с базой данных, а база данных отвечает за хранение информации о товарах, категориях и инвентаризации.

На рисунке 3 предоставлена диаграмма компонентов клиент-серверного web-приложения для учёта наличия товаров «Учётка».

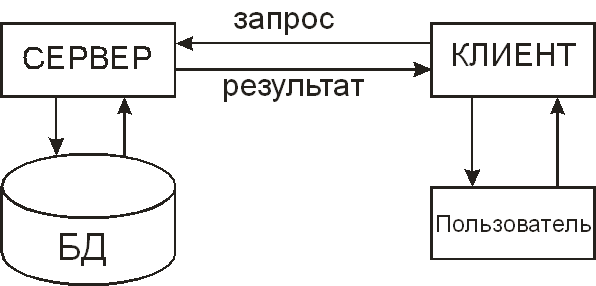


Рис. 3 – Диаграмма компонентов

# **8. Диаграмма вариантов использования**

На этой диаграмме определяются возможности пользователя. В соответствии с ней пользователь может выбрать сущность для просмотра данных, при просмотре этих данных пользователь имеет возможность добавлять, редактировать и удалять записи в БД. При редактировании пользователь выбирает товар или категорию и может редактировать любые поля, кроме уникального идентификатора. При удалении пользователь может выбрать как категорию, так и товар. При добавлении записи о продаже/покупке пользователь сначала выбирает категорию, в которой находится этот товар, потом ему на выбор всплывает список товаров, доступных в этой категории, после выбора товара он добавляет действие над этим товаром – продажа/покупка.

На рисунке 4 представлена диаграмма вариантов использования.

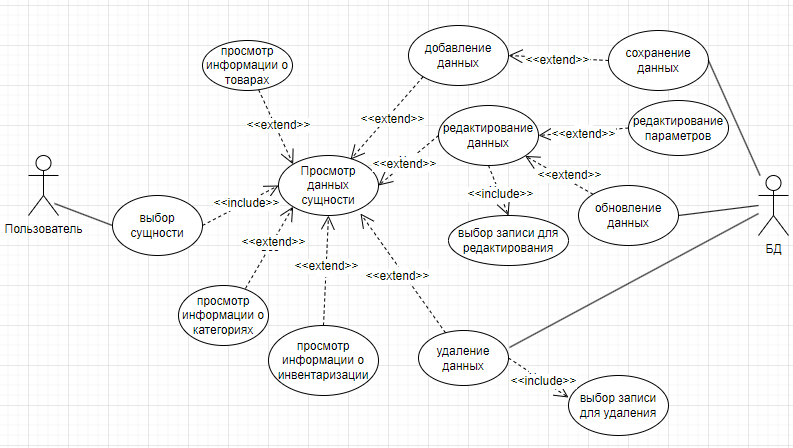


Рис. 4 – Диаграмма вариантов использования

# **9. Диаграмма последовательности**

Диаграмма последовательности – это схема, иллюстрирующая как происходит взаимодействие различных частей системы друг с другом, также она показывает порядок выполнения определенных функций в проекте.

Диаграмма представлена на рисунке 5.

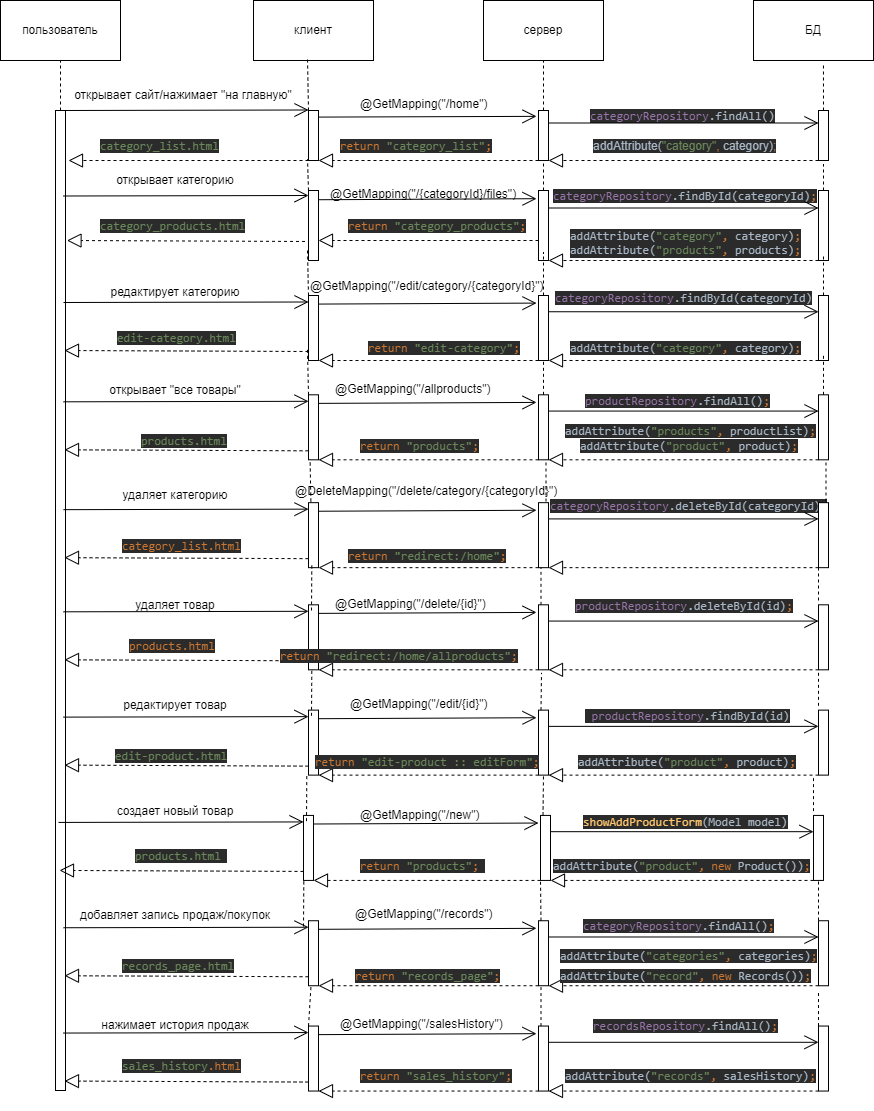


Рис. 5 – Диаграмма последовательности.

# **10. Демонстрация работы приложения**

На начальной странице пользователю предоставлен выбор категорий товаров, существующих в нашем веб-сервисе (рисунок 6)

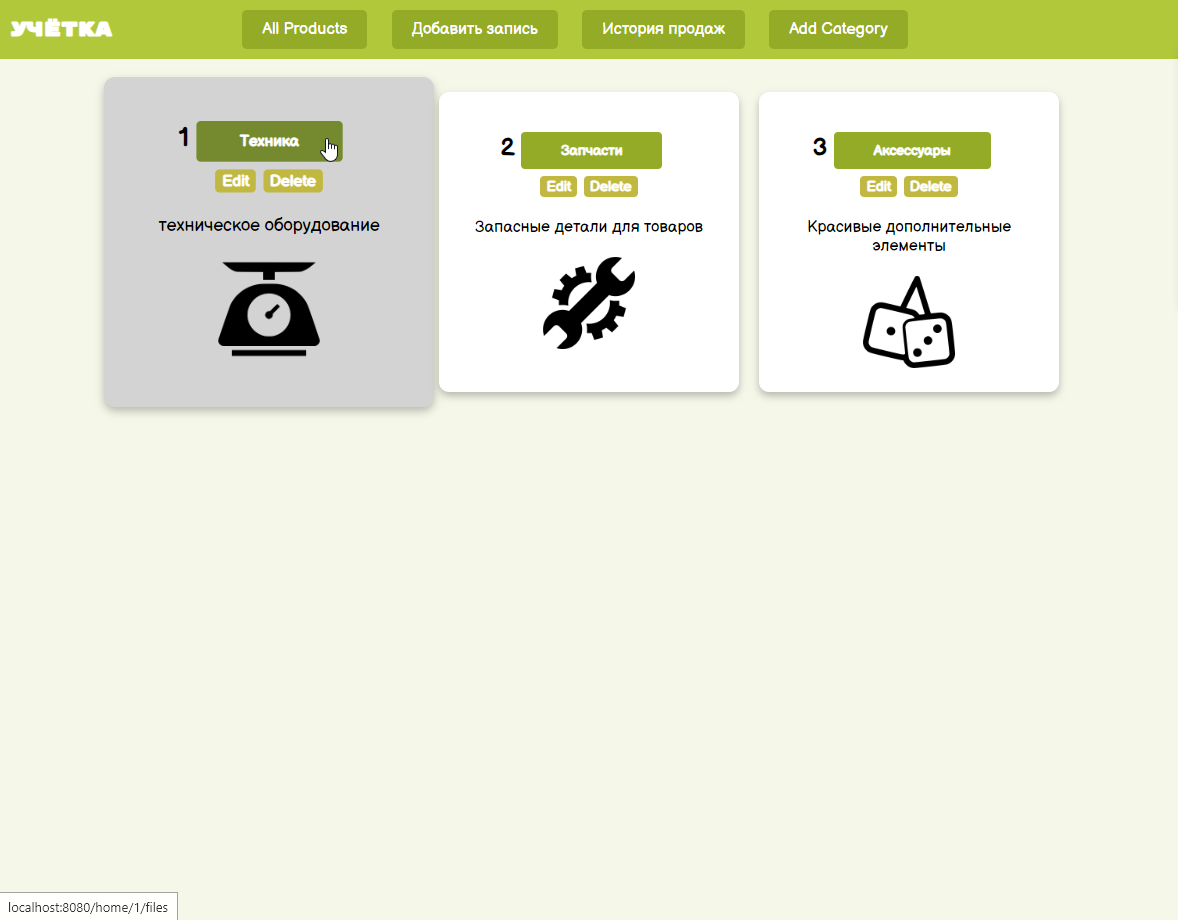


Рис. 6 – Главная страница

При нажатии на «Add category» всплывает форма для добавления новой категории. Всплывающая форма представлена на рисунке 7.

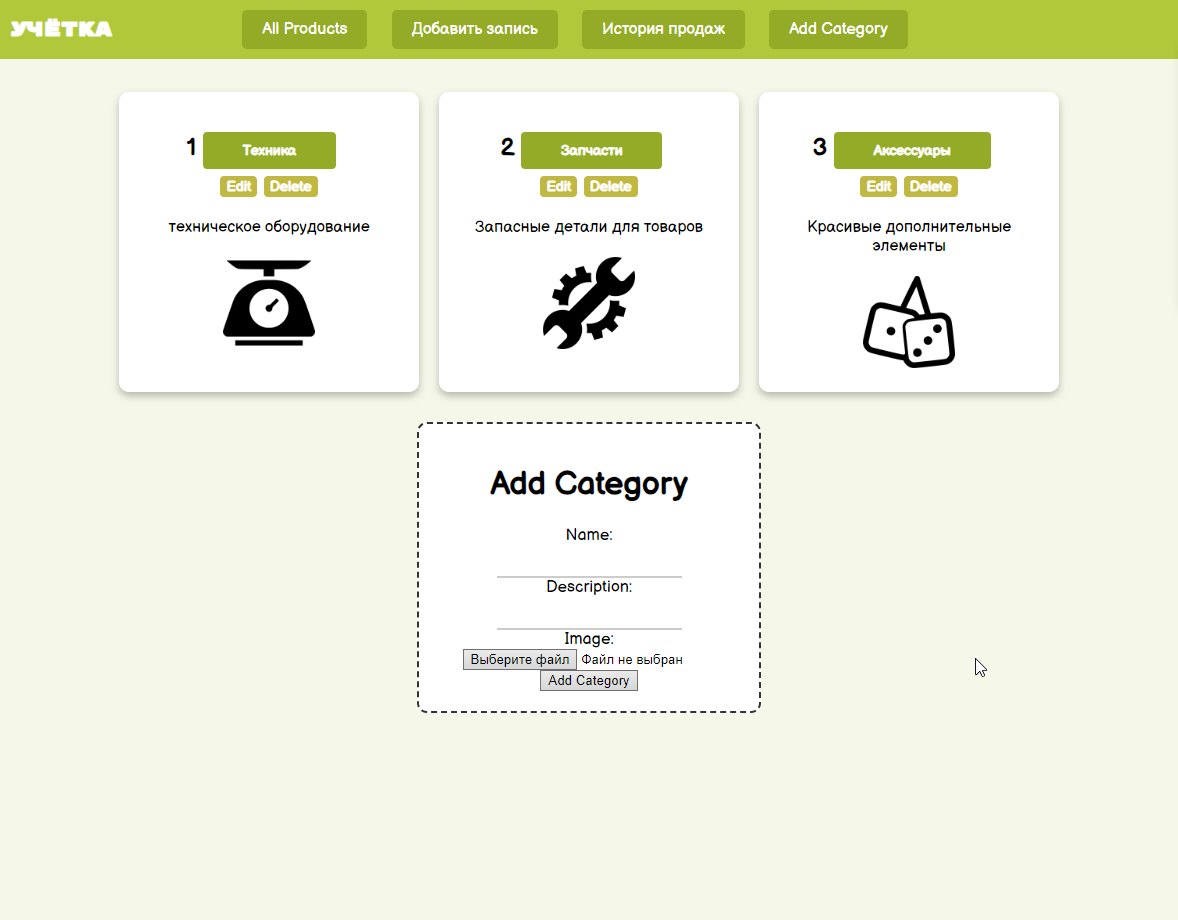


Рис. 7 – Всплывающая форма добавления новой категории.

При нажатии на любую категорию происходит переход на страницу товаров соответствующей категории. Данные страницы представлены на рисунках 8.1-8.3.

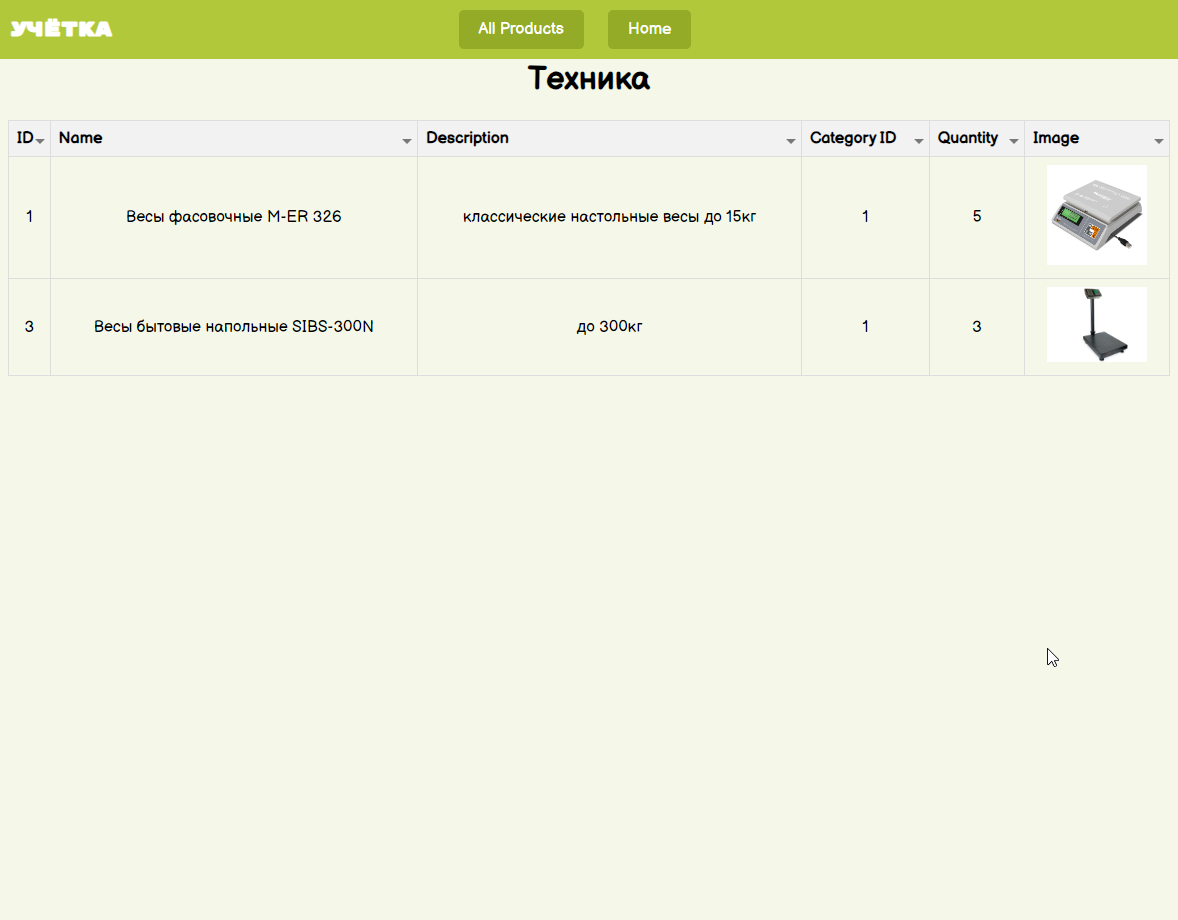


Рис. 8.1 – Страница техники.

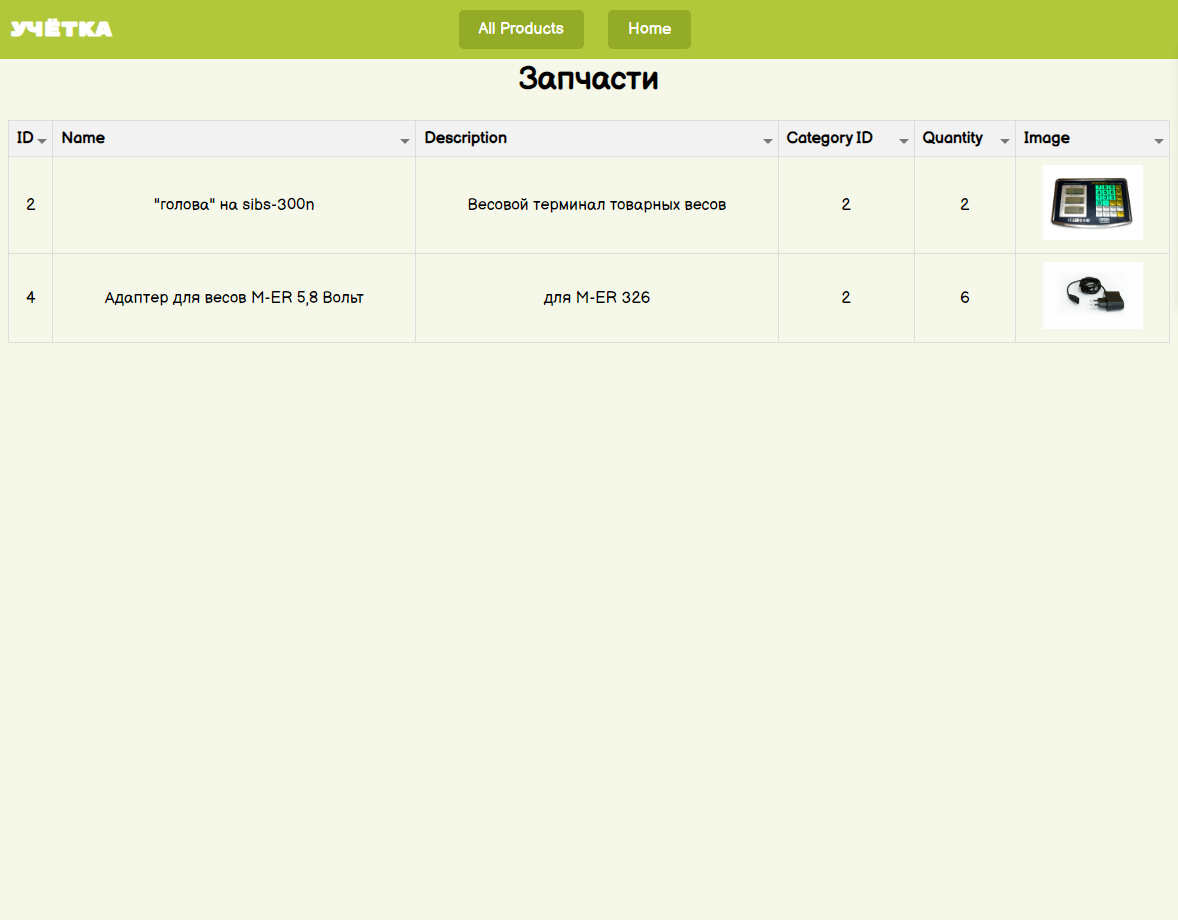


Рис. 8.2 – Страница запчастей.

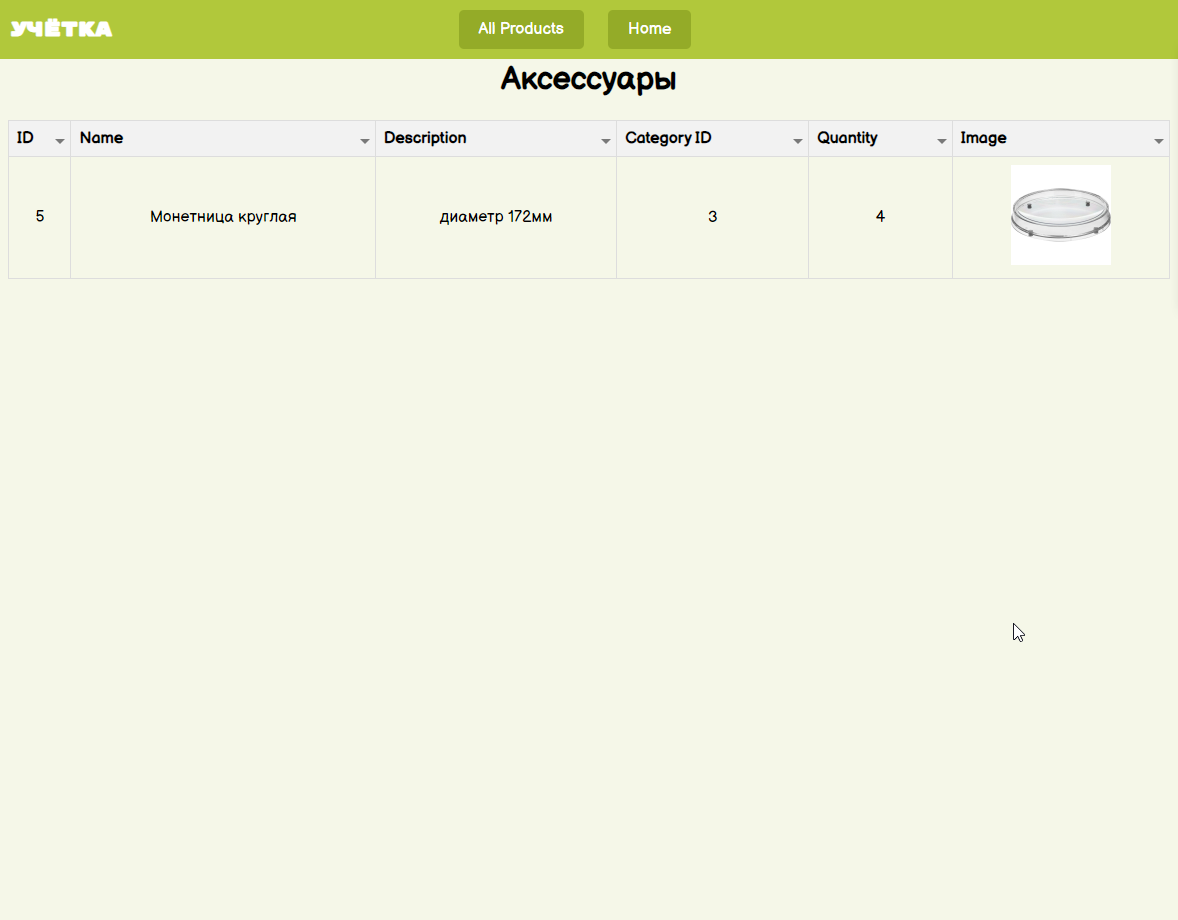


Рис. 8.3 – Страница аксессуаров.

При нажатии на кнопку «All products» происходит переход на страницу со всеми товарами, которые есть у нас в магазине. Эта страница представлена на рисунке 9.

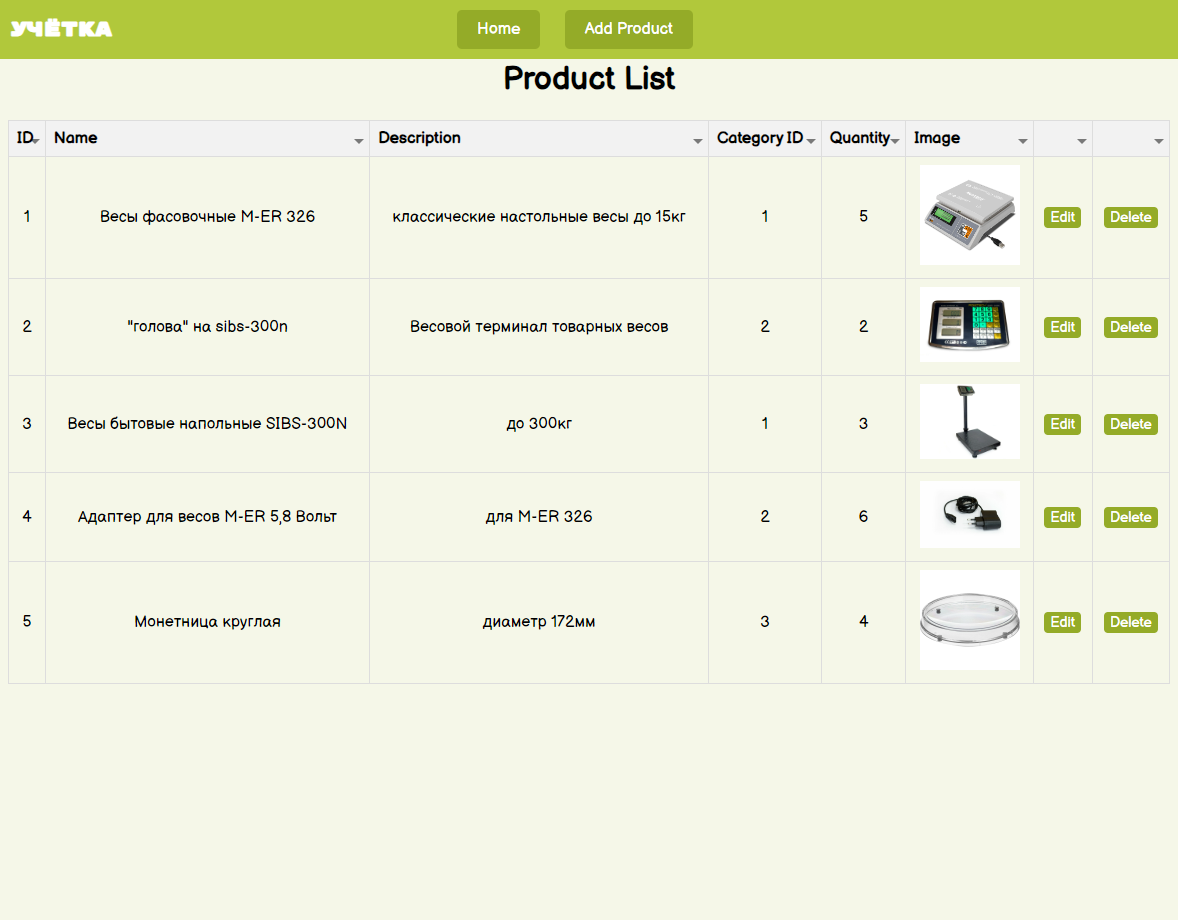


Рис. 9 – Страница всех товаров.

При нажатии на «Add Product» открывается форма для добавления нового товара (рисунок 10).

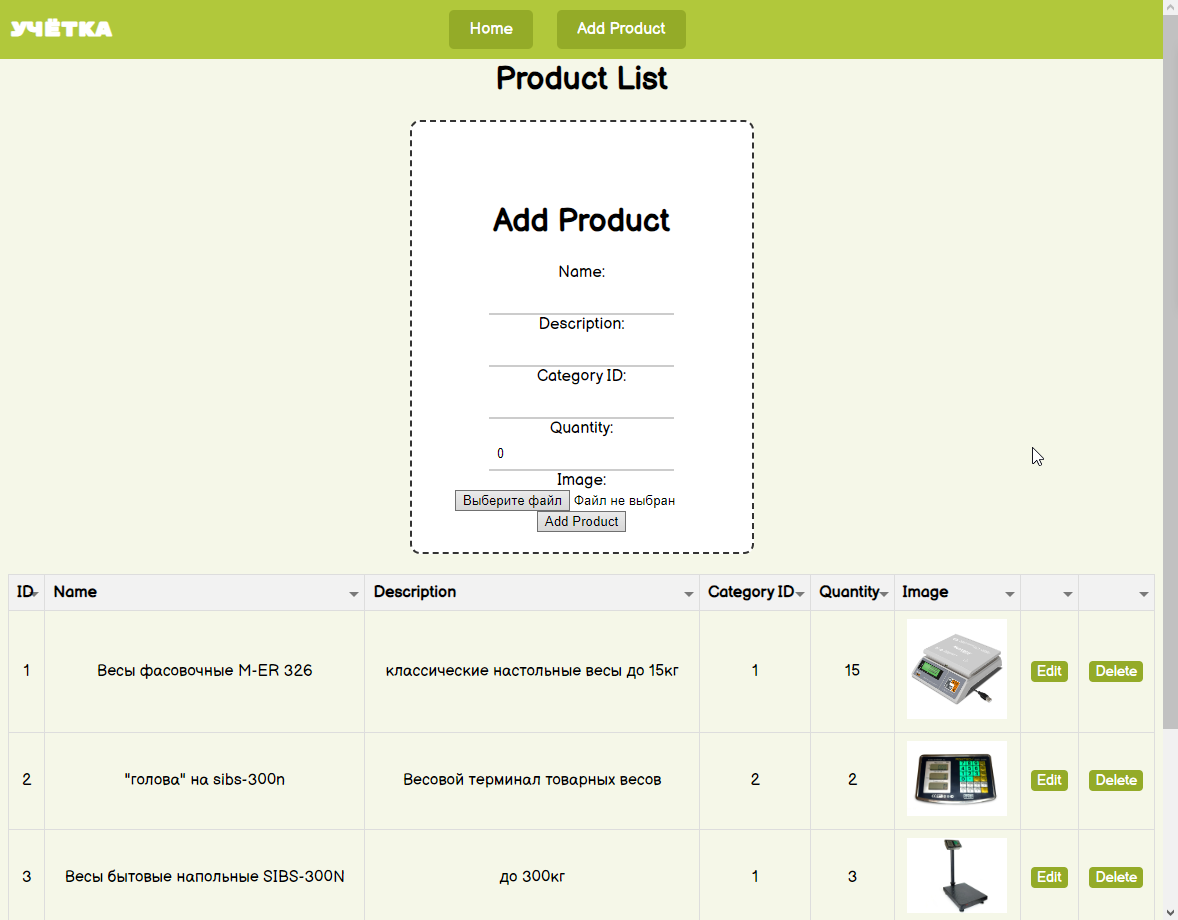


Рис. 10 – Форма добавления товара

При нажатии на кнопку «edit» сбоку от товара открывается форма для редактирования данного товара (рисунок 11).

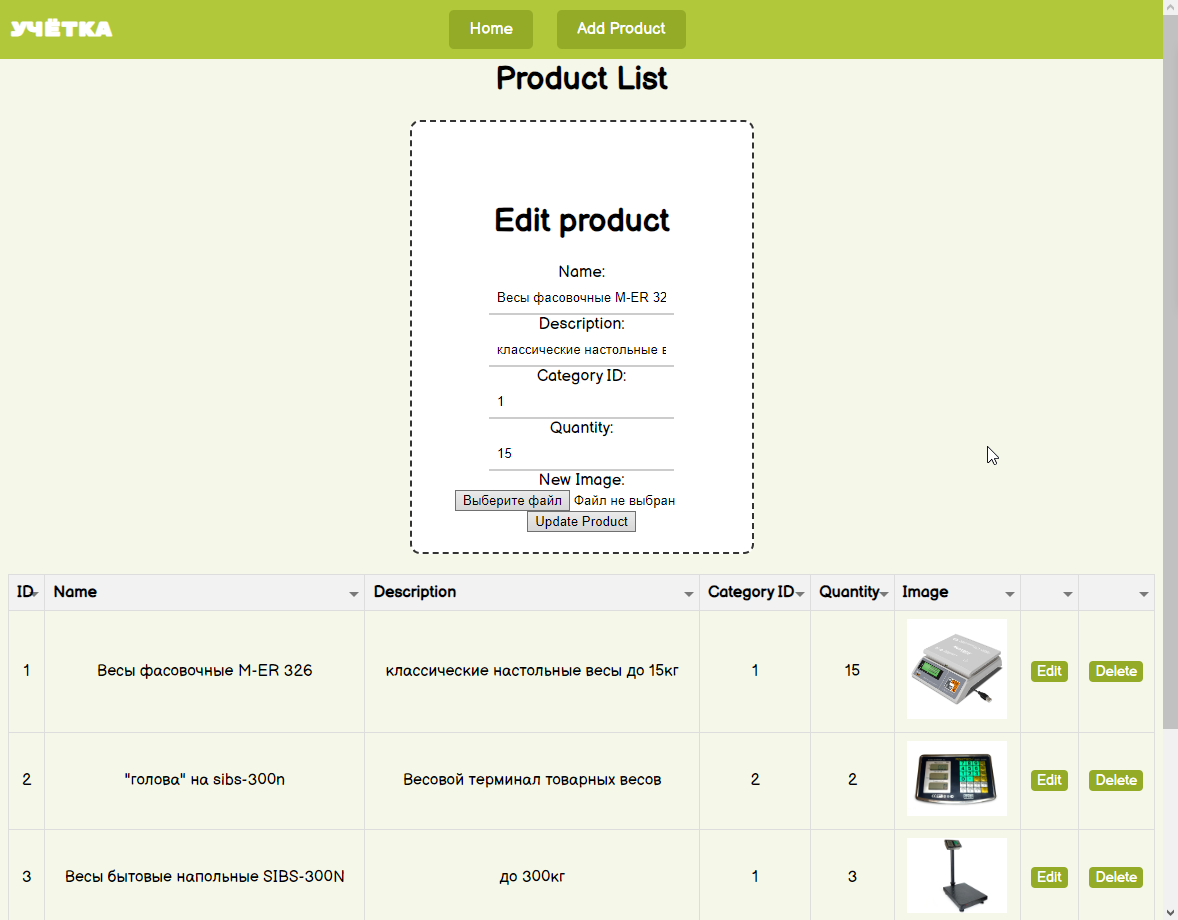


Рис. 11 – Форма редактирования товара.

При нажатии на кнопку «Добавить запись» на главной странице происходит переход на страницу с формой заполнения инвентаризации (рисунок 12).

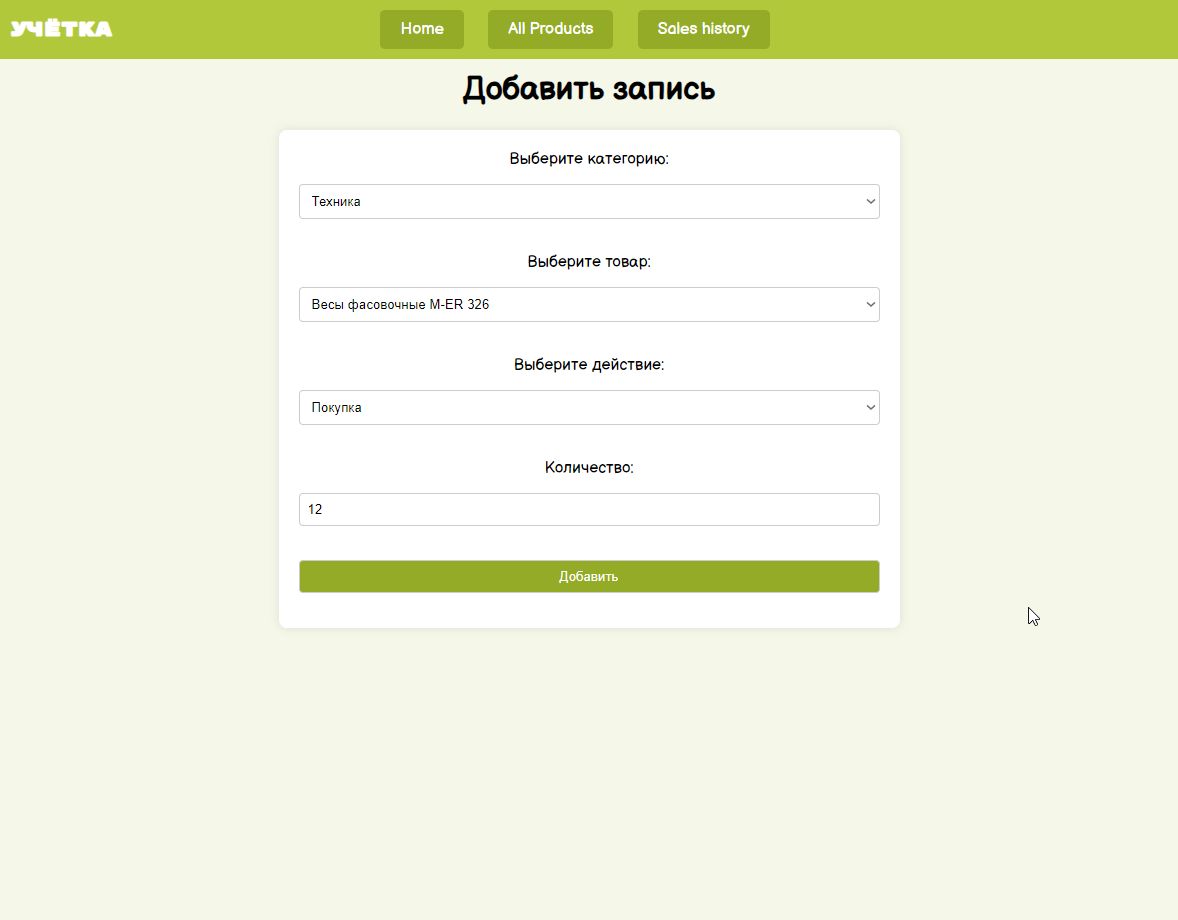


Рис. 12 – Форма заполнения инвентаризации.

При нажатии на кнопку «История продаж» на главной странице, происходит переход на страницу, где отображены истории продаж/покупок (рисунок 13).

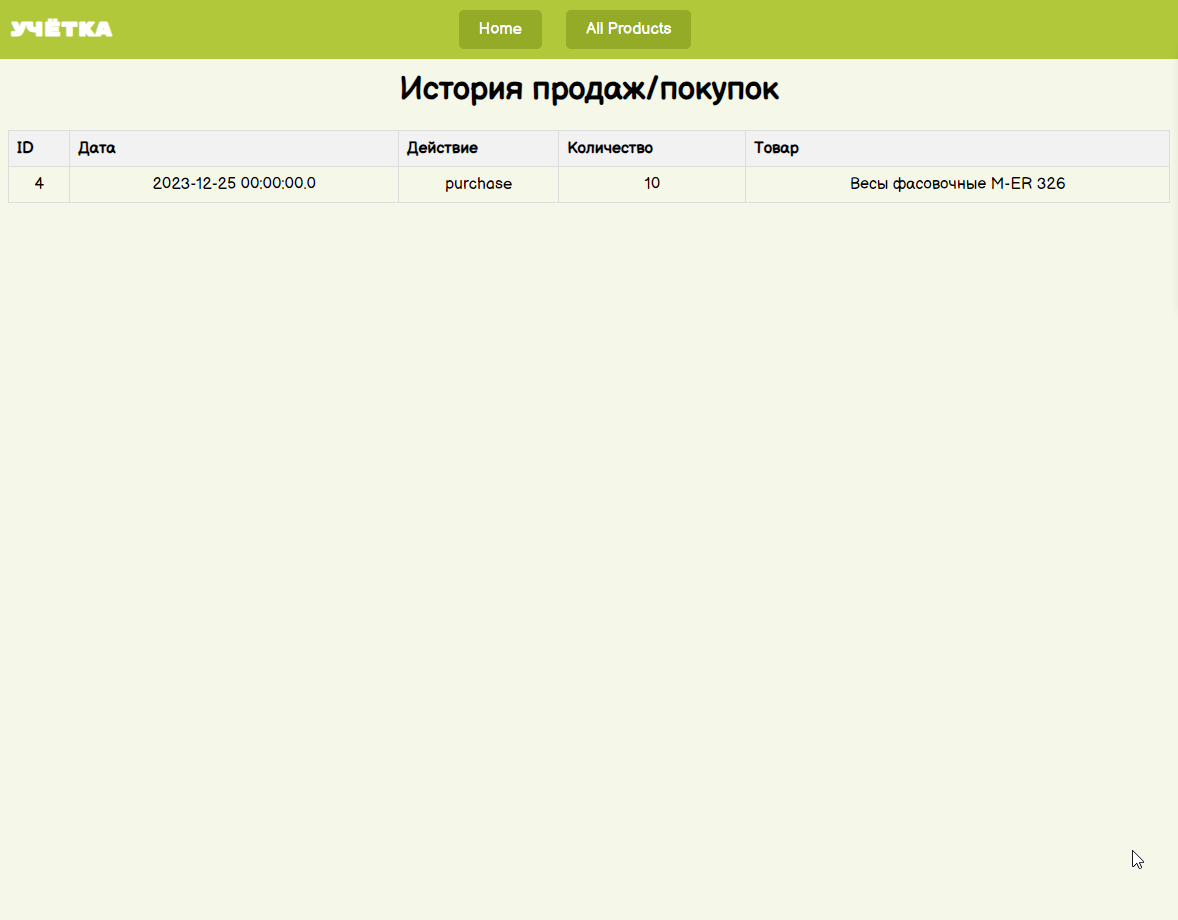


Рис. 13 – Страница отображения продаж/покупок.

При нажатии на кнопку «delete» сбоку от категории, происходит удаление нашей категории (рисунок 14).

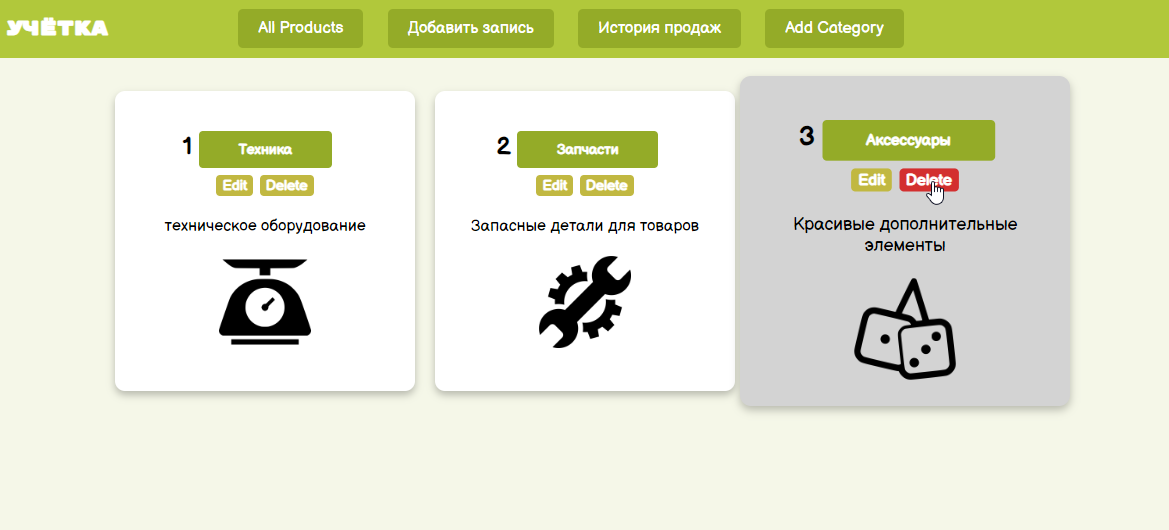


Рис. 14 – Удаление нашей категории

При нажатии на кнопку «delete» сбоку от товара, происходит удаление нашего товара (рисунок 15).

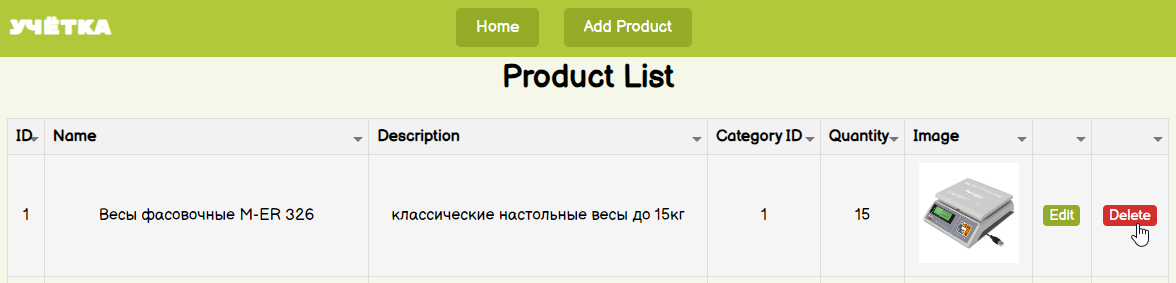


Рис. 15 – Удаление товаров

# **Список используемых источников**

1. Вишнякова, С.М. Профессиональное образование. Словарь. Ключевые понятия, термины, актуальная лексика [Электронный ресурс]/ С.М.Вишнякова – М.: НМЦ СПО, 2020.
2. Понятие информационной системы (ИС) [Электронный ресурс]. – URL: http://cde.osu.ru/demoversion/course157/text/1.5.html
3. Информационные системы. Базы данных [Электронный ресурс]. – URL: http://www.kolomna-school7-ict.narod.ru/st40401.html
4. Thymeleaf [Электронный ресурс]. – URL: <https://www.thymeleaf.org/>
5. Официальная документация Spring. URL: https://spring.io/docs
6. Бен Фрейн (Ben Frain) - "HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств", Питер, 2017, 272 стр. (ориг. название: "Responsive Web Design with HTML5 and CSS3", Packt Publishing).
7. А. Хрусталев, А. Кириченко "HTMLS + CSS3. Основы современного WEB-дизайна", Наука и Техника, 2018, 352 стр.

# **Приложение – Листинг программного кода**

CategoryController.java

package com.testim.productstest.controller;  
import …

МАКСИМУМ 10 СТРАНИЦ