

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	3
1. СБОР И АНАЛИЗ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ .....	4
1.1 Описание предметной области.....	4
2. РАЗРАБОТКА ПРИЛОЖЕНИЯ .....	7
2.1 Проектирование програмного продукта .....	7
2.2 Выбор средств и технологий разработки приложения .....	7
2.4 Структура програмного продукта.....	9
3. ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА.....	11
3.1 Тестовый план.....	11
3.2 Тест кейсы .....	14
ЗАКЛЮЧЕНИЕ.....	26
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	27

## **СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ**

ПО — Программное обеспечение;

ФИО — фамилия, имя, отчество;

БД — база данных;

СУБД — система управления базами данных;

UML — Unified Modeling Language (унифицированный язык моделирования);

IEEE — Institute of Electrical and Electronics Engineers (Институт инженеров электротехники и электроники);

JDK - Java Development Kit;

ORM - Object-Relational Mapping;

## **ВВЕДЕНИЕ**

В современном мире в связи с наличием глобальной сети Интернет, наличие ПК или ноутбука стало практически обязательным требованием для работы или отдыха.

В следствии этого возникает потребность покупки нового оборудования на замену старому или обновлении лишь части комплектующим ПК или ноутбука.

Вопрос приобретения новой техники, или её апгрейда стал особенно актуален после появления вируса COVID-19 и периода самоизоляции, во время которого обучение и работа проводятся дистанционно, соответственно даже те, у кого не было современной техники или компьютеров вовсе, были вынуждены приобретать новые модели. В период пандемии идеальным вариантом стали онлайн магазины компьютерных комплектующих.

В данной работе предполагается разработать непосредственно приложение.

Цели проекта:

1. Создание удобного приложения, которое предоставит пользователю полную информацию об интересующей продукции.
2. Создание удобной системы, которая дает возможность пользователям посмотреть продукцию, не выходя из дома.
3. Рабочий интерфейс должен быть ориентирован на самого неопытного пользователя, который будет очень прост и удобен в использовании.

Для решения поставленных задач были применены следующие основные методы исследования: теоретические (исследование предметной области), эмпирические (анализ качества разработанного комплекса, методы построения дизайна сайта, выбор средств разработки).

Объектом исследования является приложение магазина компьютерных комплектующих.

Предметом исследования является разработка приложения «Магазин компьютерных комплектующих».

# **1. СБОР И АНАЛИЗ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ПРОДУКТУ**

## **1.1 Описание предметной области**

Предметной областью для данной работы является разработка приложения.

Разработанное веб-приложение в выполненной работе имеет следующее название: «Estore».

Полное наименование - «Приложение на тему «Магазин компьютерных комплектующих»».

В настоящее время остро стоит вопрос о покупке новой компьютерной техники в связи с постоянным переходом на цифровые инструменты для работы и учёбы и постепенным отказом от бумажных носителей информации.

Разработанное веб-приложение представляет собой систему ведения информации о имеющихся продуктах. Он позволяет клиенту просматривать информацию о компьютерных компонентах и совершать покупки. А администратору возможность добавлять новые продукты и отслеживать историю заказов.

При разработке проекта для обеспечения его максимальной эффективности очень важно учитывать все особенности информационной среды, в которой будет происходить эксплуатация. Информационная среда приложения представлена пользователями.

Для сравнительного анализа были выбраны такие интернет-магазины компьютерных комплектующих как «Регард», режим доступа: <https://www.regard.ru> и «Ситилинк», режим доступа: <https://www.citilink.ru>.

«Регард» имеет удобную навигацию и интерфейс, а также дизайн, где преобладают красный и белый цвета. Помимо этого сайт имеет возможность покупки товаров без регистрации, что упрощает и ускоряет процесс

формирования заказа для тех, кто впервые делает покупки на сайте, либо не имеет желания регистрироваться. Интерфейс сайта изображён на рисунке 1.1.

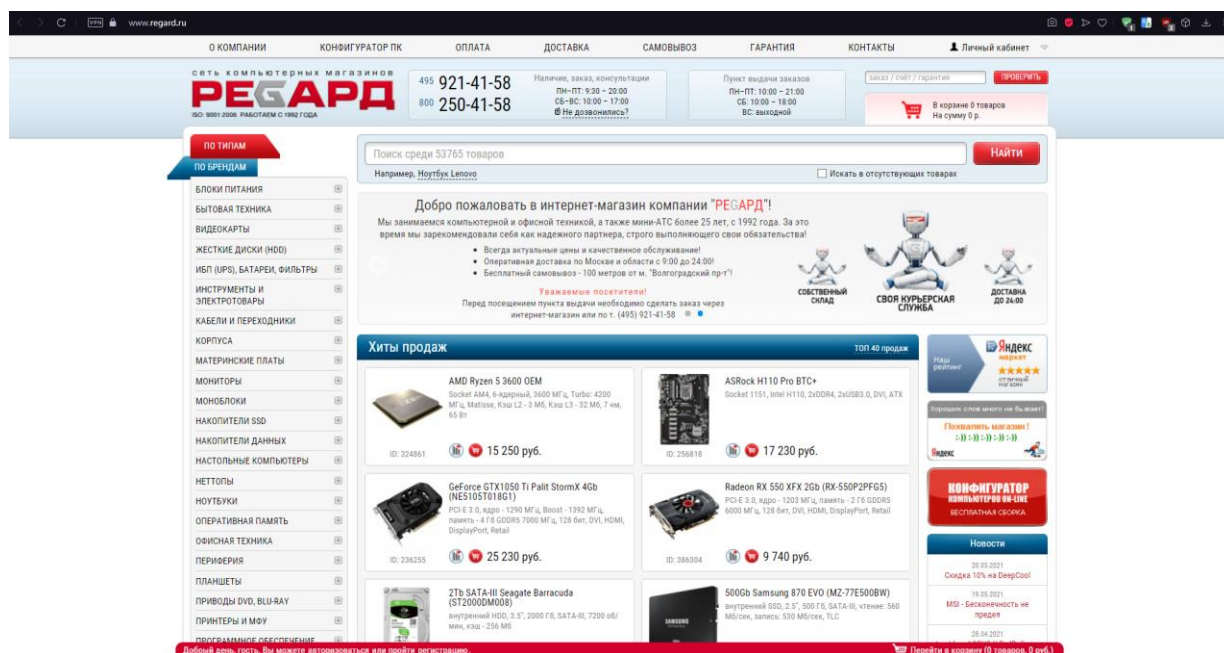


Рисунок 1.1 – Интерфейс онлайн-магазина «Регард»

«Ситилинк» же обладает минималистичным дизайном и минимумом ненужных и отвлекающих элементов, что облегчает адаптацию и фокусировку на необходимом товаре при посещении сайта. Интерфейс интернет-ресурса изображён на рисунке 1.2.

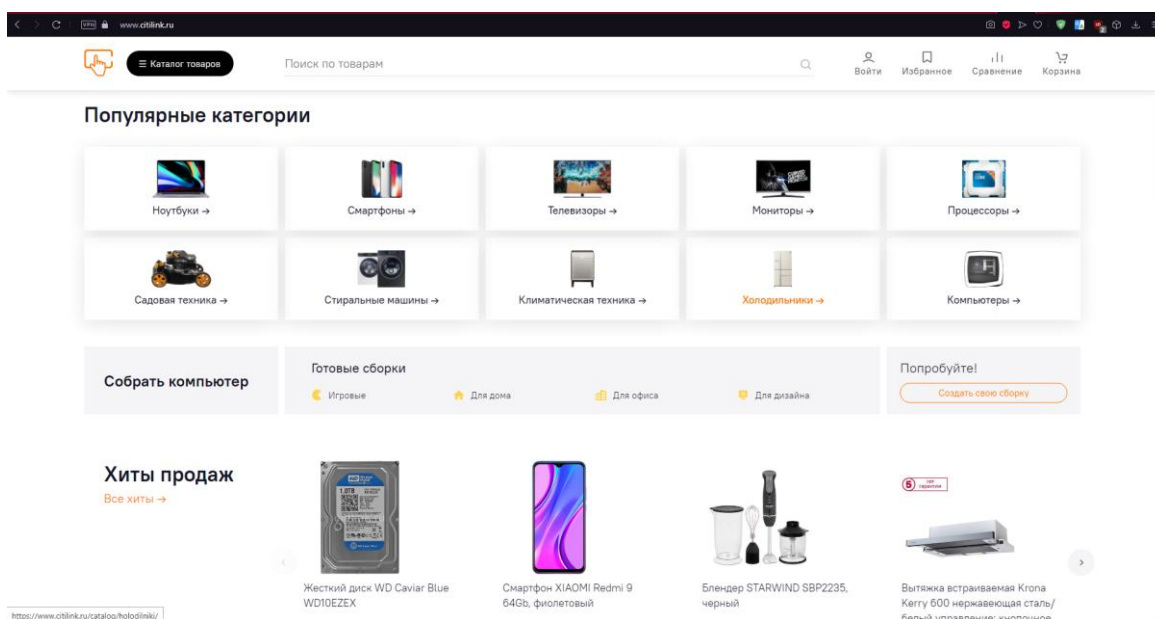


Рисунок 1.2 – Интерфейс онлайн-магазина «Ситилинк»

На рисунке изображён интерфейс приложения (Рис. 1.3)

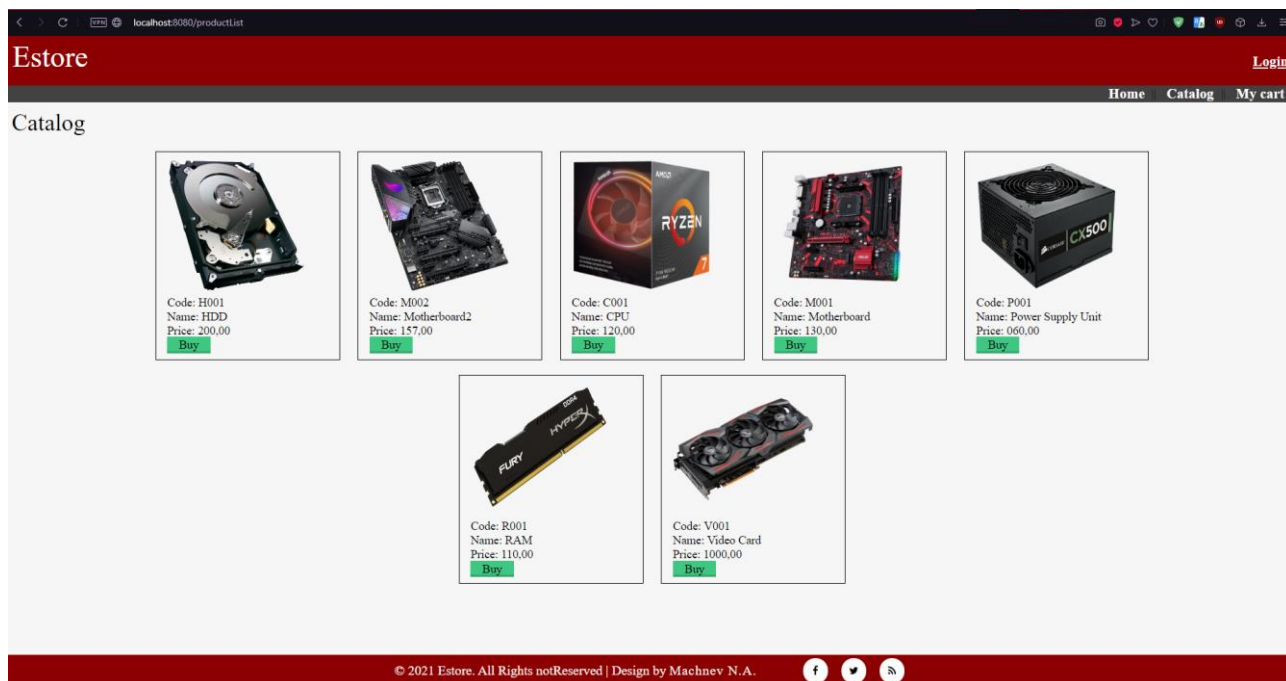


Рис. 1.3 – Интерфейс приложения

## Выводы к разделу 1

Проведя анализ, можно сделать вывод, что в эпоху всеобщей цифровизации и появившейся внезапно пандемии, спрос на электронную технику, а в частности на ПК, ноутбуки и их комплектующие сильно возрос.

И именно магазины компьютерных комплектующих смогут покрыть этот спрос.

В ходе анализа предметной области были сформированы требования к приложению:

- Создание базы данных для товаров, заказов и пользователей;
- Осуществлении работы с использованием технологии Spring MVC и ее составляющих, Java;
- Осуществление удобной автоматической сборки проекта с помощью Maven;
- Веб-приложение должно обладать информативностью и минималистичным дизайном;

## **2. РАЗРАБОТКА ПРИЛОЖЕНИЯ**

### **2.1 Проектирование программного продукта**

После формирования списка общих требований в анализе предметной области необходимо сформулировать список требований, которые будут отражать видение самого проекта и его функционала.

Список основных требований (функции и технологии, которые должны быть использованы при работе с проектом):

1. Использовать фреймворк Spring MVC как основу проекта;
2. Веб-приложение должно обладать интерактивностью поведения;
3. Обеспечение безопасности приложения с помощью RESTful web-сервисов;
4. Для работы с БД использовать технологию Spring DAO;
5. Использование фреймворка для автоматизации сборки проектов Maven;
6. Подключение к системе управления проектами и версиями кода GitHub;
7. В приложении необходимо наличие интуитивно понятного интерфейса;
8. Разработка проекта в интегрированной среде разработки программного обеспечения IntelliJ IDEA;
9. В приложении должна присутствовать удобная навигация;

### **2.2 Выбор средств и технологий разработки приложения**

В проектировании использовались следующие технологии и веб-средства для достижения поставленной цели:

JDK 14 — бесплатный пакет для разработчика приложений и апплетов, написанных на языке Java.

В качестве основного фреймворка использовался Spring MVC.

Restful Web Service — сервис, используемый для безопасного для клиента соединения с сайтом.

MySQL — система управления реляционными базами данных.

ORM — технология, используемая для отображения объектов в структуры реляционных баз данных.

Thymeleaf — современный серверный механизм Java-шаблонов для веб- и автономных сред, способный обрабатывать HTML, XML, JavaScript, CSS и даже простой текст.

Spring Security - фреймворк, предоставляющий механизмы построения систем аутентификации и авторизации, а также другие возможности обеспечения безопасности для промышленных приложений, созданных с помощью Spring Framework.

Apache Maven — Фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML.

GitHub - это система управления проектами и версиями кода, а также платформа социальных сетей, созданная для разработчиков.

IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python.

HTML5 — язык для структурирования и представления содержимого всемирной паутины. Это пятая версия HTML.

CSS3 — Формальный язык описания внешнего вида документа, написанного с использованием языка разметки

Opera — веб-браузер и пакет прикладных программ для работы в Интернете, выпускаемый компанией Opera Software.



## 2.3 Структура программного продукта

Веб-приложение состоит из двух частей: клиентской и серверной.

Пользователь взаимодействует с веб-сервером при помощи браузера.

Клиентская часть реализует пользовательский интерфейс, формирует запросы к серверу и обрабатывает ответы от него.

Серверная часть получает запрос от клиента, выполняет вычисления, после этого формирует веб-страницу и отправляет её клиенту по сети с использованием протокола HTTP.

На стороне клиента для реализации GUI используются такие языки как HTML, CSS и JavaScript, а на стороне сервера Java.

На рисунке 2.1 изображена схема структуры программного продукта.

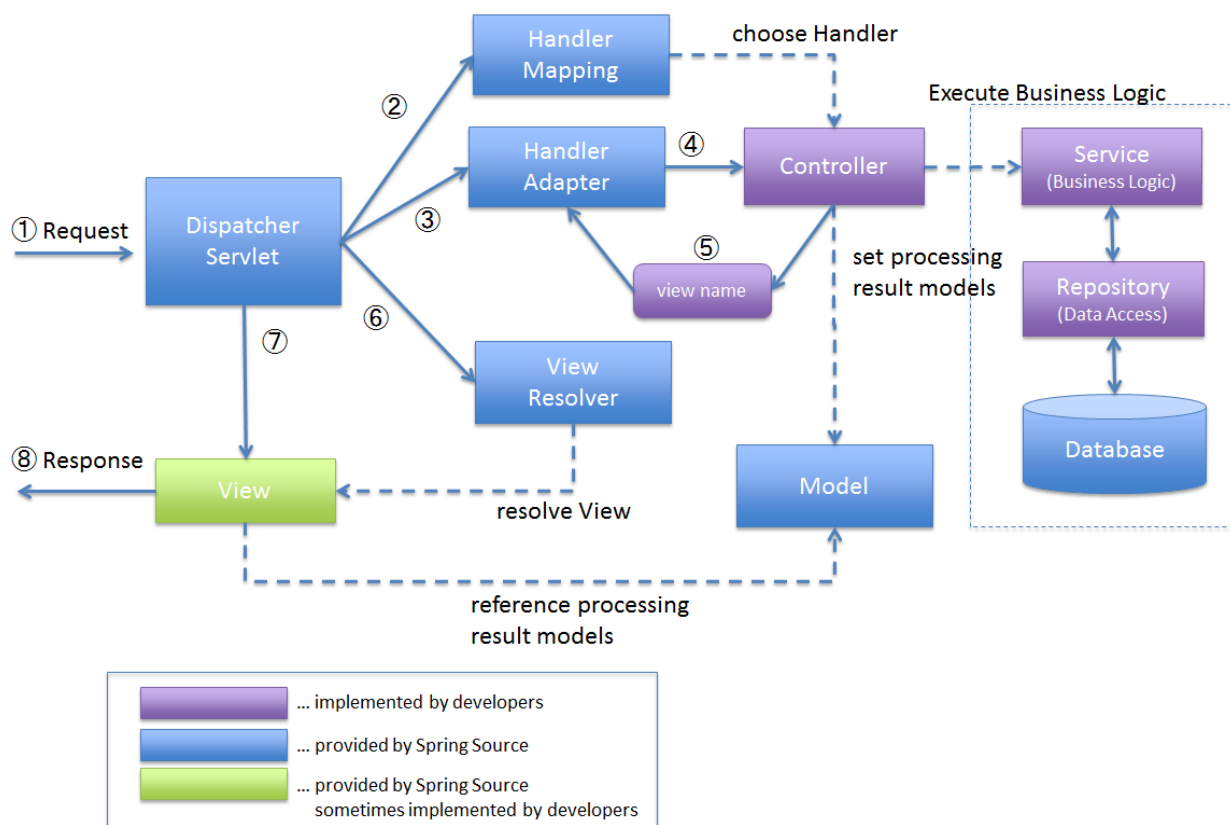


Рисунок 2.1 – Схема структуры программного продукта

## **Выводы к разделу 2**

В результате анализа были выбраны наиболее подходящие технологии, с помощью которых был разработан программный продукт.

Также были проанализированы требования к программному продукту и были определены функции, которыми должно обладать ПО.

Был составлен список дополнительных требований к веб-приложению, которые сделают его более удобным в использовании.

Было разработано веб-приложение, отвечающее всем представленным требованиям.

### **3. ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА**

В этом разделе описывается тестирование программного продукта.

#### **3.1 Тестовый план**

Тестовый план составляется по стандарту «IEEE 829» с удалением нескольких пунктов из-за ограниченности проводимого тестирования.

##### **3.1.1 Test Plan Identifier (идентификатор)**

Тестовый план нумеруется как «id\_1», уровень тестового плана - «plan\_level\_1», уровень программного обеспечения - «websoftware\_1». План является мастер-планом («Master Plan»).

Номер ревизии плана: 1.

Контактная информация главного составителя плана тестирования:

ФИО: Мачнев Никита Антонович.

Email: Pistolero01@mail.ru.

Номер телефона: +7 (925) 422-31-01.

##### **3.1.2 Цель**

Целью плана является создание условий тестирования функций ПО «веб-приложения «Магазин компьютерных комплектующих»», улучшение работоспособности системы, посредством уменьшения вероятности отказа работы основных модулей программы.

Тестирование функций и возможностей приложения необходимо для повышения удовлетворения пользователей сайта при его использовании, а также уменьшение количества ошибок ПО в процессе эксплуатации.

##### **3.1.3 Области, подвергаемые тестированию**

- 1) Авторизация пользователей на сайте;
- 2) Добавление, удаление и изменение товаров администраторами;

- 3) Добавление товара в корзину и оформление заказ с авторизацией и без неё;
- 4) Просмотр списка заказов и подробной информации о них администраторами;

#### **3.1.4 Области, не подвергаемые тестированию**

- 1) Программы и технологии с помощью которых разрабатывается веб-приложение;
- 2) Бизнес-аналитика приложения;
- 3) Сетевая безопасность сервера;

#### **3.1.5 Тестовая стратегия и подходы**

Тестирование приложения будет проходить методом «белого ящика».

Уровни тестирования:

- 1) Системное/интеграционное тестирование

Проводится программистом. Системное/интеграционное тестирование происходит после того, как критические дефекты исправлены. После этого вида тестирования выполняется приёмочное тестирование.

- 2) Приемочное тестирование

Проводится конечными пользователями (студентами и преподавателями).

#### **3.1.6 Критерии**

Необходимо, чтобы любой пользователь имел возможность просмотреть список товаров и делать заказ без авторизации.

После этого авторизированный пользователь с ролью «Администратор» мог бы просмотреть список заказов и подробную информацию о них.

Также пользователь с ролью «Администратор» должен иметь возможность изменять, удалять и добавлять товары на сайт.

Для каждой тестируемой метрики её показатели «выполнение тест-кейсов» и «успешное прохождение тест-кейсов» должны быть больше 60 процентов и меньше 80 процентов соответственно. В этом случае тестирование продолжается.

### **3.1.7 Ресурсы**

- Операционная система: Windows 10 Pro
- Установленная СУБД MySQL версии 8.0
- Установленный Spring Framework

### **3.1.8 Роли и ответственность**

Программист ответственен за выполнение системного/интеграционного тестирования.

Конечные пользователи ответственны за приёмочное тестирование.

### **3.1.9 Метрики**

- Покрытие требований тест-кейсами – процентное отношение количества покрытых тест-кейсами требований к общему количеству требований.
- Успешное прохождение тест-кейсов – процентное отношение количества успешно выполненных тест-кейсов к общему количеству выполненных тест-кейсов.
- Выполнение тест-кейсов – процентное отношение количества выполненных тест-кейсов к количеству тест-кейсов, запланированных к выполнению.

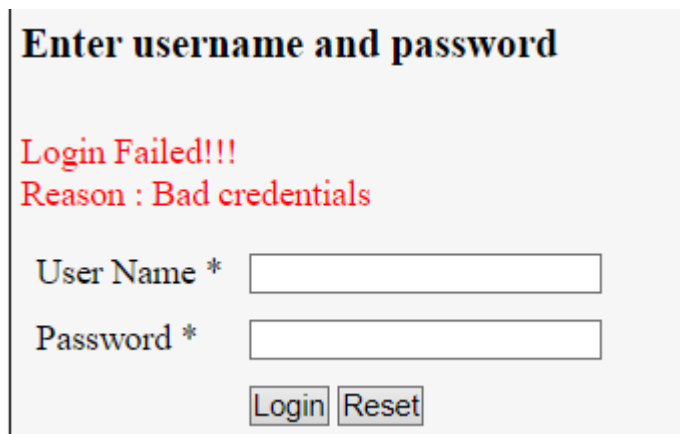
### 3.2 Тест кейсы

В таблице 3.1 представлен сформированный тест-кейс для тестирования авторизации пользователя в системе.

Таблица 3.1 – Тест-кейс для тестирования авторизации на сайте.

ID: 1	Название: Авторизация на сайте		
R_1	Мачнев Н.А.	Высокий приоритет	Модуль: Форма авторизации
Приготовления на начало тестирования: 1. Перейти на сайт проекта и нажать на кнопку «Login» в верхней панели.			
Шаги: 1. Нажать на поля ввода и ввести туда корректные требуемые данные (логин и пароль). 2. Нажать на кнопку «Login» в форме для входа.		Ожидаемый результат: 1. Вывод сообщение об ошибке и её причины в случае неправильного ввода. 2. Если авторизация прошла успешно будет показаны информация о пользователе, такие как имя пользователя в системе и его роль. В верхней панели вместо кнопки «Login» появится имя пользователя и кнопка «Logout» 3. В случае авторизации пользователя с ролью «Администратор» в меню сайта добавится страница со списком заказов и откроется возможность редактировать товары.	
Статус «Разработано»		Дата выполнения 15.05.21	Ссылка на дефекты:

Тест ожидаемого результата №1 изображен на рисунке 3.1.



The screenshot shows a web form titled "Enter username and password". Below the title, there is a red error message: "Login Failed!!! Reason : Bad credentials". The form contains two input fields: "User Name \*" and "Password \*". Below the input fields are two buttons: "Login" and "Reset".

Рисунок 3.1 – Результат теста ожидаемого результата №1

Тест ожидаемых результатов №2 и №3 изображен на рисунке 3.2.

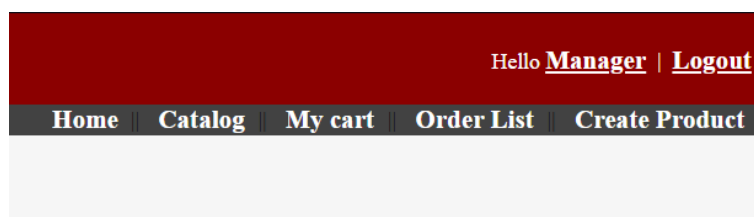


Рисунок 3.2 - Результат теста ожидаемых результатов №2 и №3

В таблице 3.2 представлен сформированный тест-кейс для тестирования корзины покупателя.

Таблица 3.2 – Тест-кейс для тестирования корзины.

ID: 2	Название: Корзина		
R_2	Мачнев Н.А.	Высокий приоритет	Модуль: Корзина
Приготовления на начало тестирования:			
1. Перейти на сайт проекта и нажать на кнопку «Catalog» в меню сайта.			

<p>Шаги:</p> <ol style="list-style-type: none"> <li>1. Выбрать товар и нажать кнопку «Buy» в окне товара.</li> <li>2. Удалить товар из списка, нажав кнопку «Delete» в окне товара</li> <li>3. Изменить количество товара в поле «Quantity» и нажать кнопку «Update Quantity» в нижнем меню.</li> <li>4. Нажать кнопку «Continue Buy» чтобы вернуться обратно в каталог.</li> </ol>	<p>Ожидаемый результат:</p> <ol style="list-style-type: none"> <li>1. После нажатия кнопки «Buy» пользователя перенесёт на страницу корзины.</li> <li>2. После нажатия кнопки «Delete» товар удалится из корзины.</li> <li>3. В случае нажатия кнопки «Update Quantity» итоговая сумма товара измениться если было изменено количество товара в поле «Quantity».</li> <li>4. После нажатия кнопки «Continue Buy» пользователь вернётся на страницу каталога.</li> </ol>	
Статус «Разработано»	Дата выполнения 15.05.21	Ссылка на дефекты:

Тест ожидаемого результата №1 изображена на рисунке 3.3.

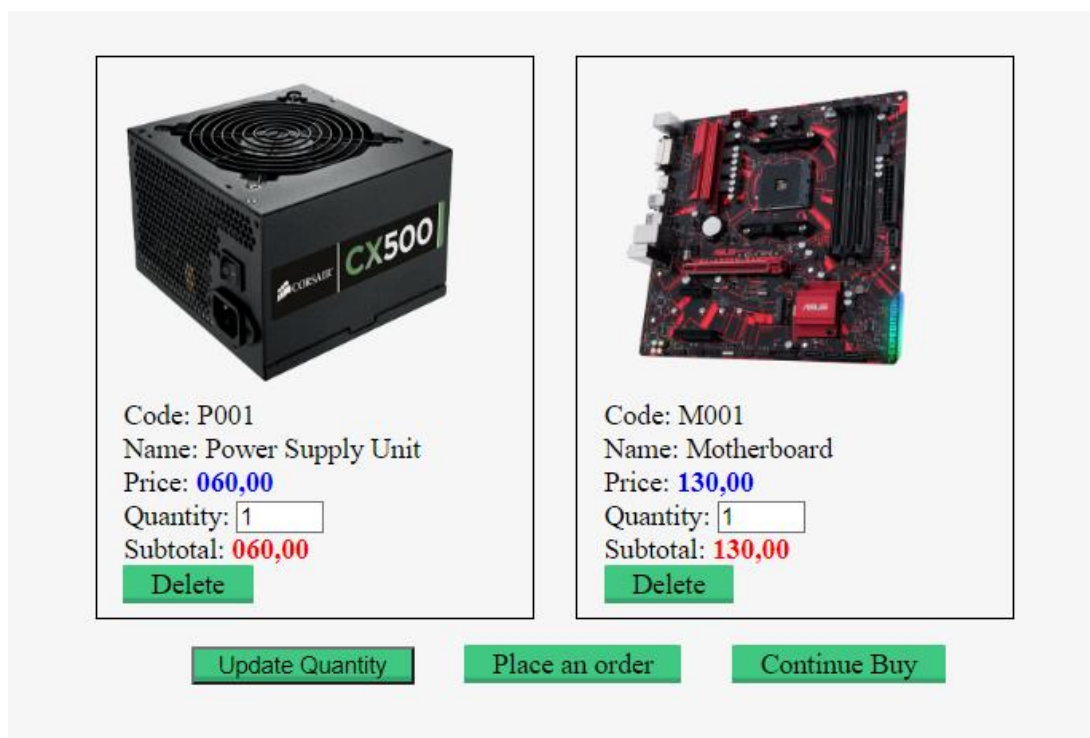


Рисунок 3.3 - Результат теста ожидаемого результата №1



Тест ожидаемого результата №2 изображён на рисунке 3.4.



Рисунок 3.4 - Результат теста ожидаемого результата №2

Тест ожидаемого результата №3 изображён на рисунке 3.5.



Рисунок 3.5 - Результат теста ожидаемого результата №3

В таблице 3.3 представлен сформированный тест-кейс для тестирования функции размещения заказа.

Таблица 3.3 – Тест-кейс для тестирования размещения заказа.

ID: 3	Название: Размещение заказа		
R_3	Мачнев Н.А.	Высокий приоритет	Модуль: Форма размещения заказа
<p>Приготовления на начало тестирования:</p> <p>1. Перейти в каталог товаров на сайте и выбрать необходимые товары.</p>			
<p>Шаги:</p> <ol style="list-style-type: none"> <li>1. Находясь на странице корзины покупок нажать кнопку «Place an order»;</li> <li>2. Заполнить все поля, отмеченные знаком «*»;</li> <li>3. Нажать кнопку «Submit», чтобы подтвердить введенные данные;</li> <li>4. Если необходимо изменить состав заказа нажать кнопку «Edit Cart»;</li> <li>5. Если необходимо изменить личную информацию нажать кнопку «Edit Customer Info»;</li> <li>6. Для создания заказа нажать кнопку «Send».</li> </ol>		<p>Ожидаемый результат:</p> <ol style="list-style-type: none"> <li>1. После нажатия кнопки «Place an order» пользователя перенесёт на страницу ввода личных данных.</li> <li>2. После нажатия кнопки «Submit» откроется страница с введенными данными и товарами.</li> <li>3. В случае неправильного ввода данных либо отправки пустого поля выведется сообщение об ошибке.</li> <li>4. При нажатии кнопки «Edit Cart» откроется страница корзины пользователя, где появится возможность изменить количество товара.</li> <li>5. При нажатии кнопки «Edit Customer Info» откроется страница с формой для ввода личных данных.</li> <li>6. После нажатия кнопки «Send» в базе данных появится запись в таблицу</li> </ol>	

	«Orders»(заказ будет создан) и появится сообщение об успешном заказе.	
Статус «Разработано»	Дата выполнения 15.05.21	Ссылка на дефекты:

Скриншот тестовой корзины изображён на рисунке 3.6.

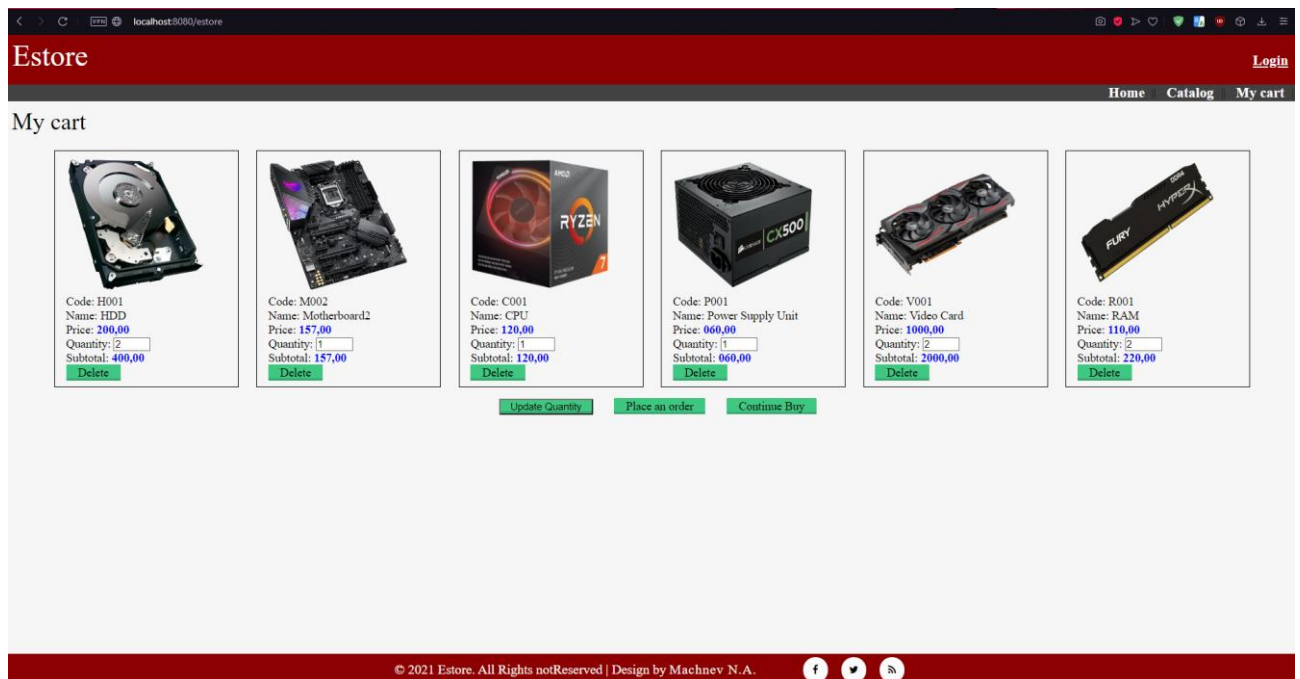


Рисунок 3.6 – Скриншот тестовой корзины

Тест ожидаемого результата №1 изображён на рисунке 3.7.

**Estore**

## Enter Customer Information

Name \*

Email \*

Phone \*

Address \*

Рисунок 3.7 – Результат теста ожидаемого результата №1

Тест ожидаемого результата №2 изображён на рисунке 3.8.

**Estore** [Login](#)

[Home](#) [Catalog](#) [My cart](#)

### Confirmation

**Customer Information:**  
Name: Customer2  
Email: customer2@mail.ru  
Phone: +89161234569  
Address: Slovakia

**Cart Summary:**  
Quantity: 9  
Total: **2957,00\$**

[Edit Cart](#) [Edit Customer Info](#) [Send](#)

Code	Name	Price	Quantity	Subtotal
H001	HDD	200,00	2	400,00\$
M002	Motherboard2	157,00	1	157,00\$
C001	CPU	120,00	1	120,00\$
P001	Power Supply Unit	660,00	1	660,00\$
V001	Video Card	1000,00	2	2000,00\$
R001	RAM	110,00	2	220,00\$

© 2021 Estore. All Rights notReserved | Design by Machnev N.A.

Рисунок 3.8 – Результат теста ожидаемого результата №2

Тест ожидаемого результата №3 изображён на рисунке 3.9.

**Estore**

## Enter Customer Information

Name \*  *Name is required*

Email \*  *Email is required*  
*Email is not valid*

Phone \*  *Phone is required*

Address \*  *Address is required*

Рисунок 3.9 – Результат теста ожидаемого результата №3

Тест ожидаемого результата №6 изображён на рисунке 3.10.

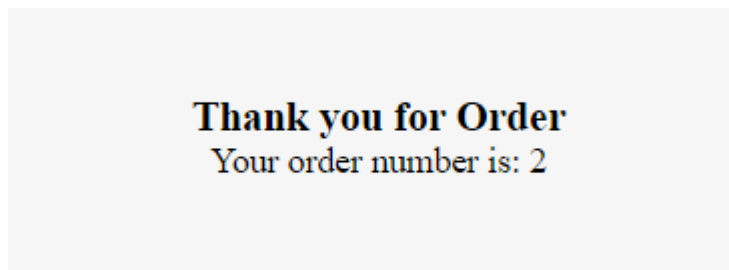


Рисунок 3.10 - Результат теста ожидаемого результата №6

В таблице 3.4 представлен сформированный тест-кейс для тестирования возможностей пользователя с ролью «Администратор».

Таблица 3.4 – Тест-кейс для тестирования пользователя «Администратор».

ID: 4	Название: Администратор		
R_4	Мачнев Н.А.	Высокий приоритет	Модуль: Форма размещения заказа

<p>Приготовления на начало тестирования:</p> <p>1. Перейти на сайт и нажать кнопку «Login».</p>	
<p>Шаги:</p> <ol style="list-style-type: none"> <li>1. Ввести данные аккаунта администратора;</li> <li>2. Перейти в каталог и изменить любой товар нажав на кнопку «Edit Product» в окне товара;</li> <li>3. Нажать кнопку «Create Product» в меню сайта;</li> <li>4. Ввести данные нового товара и нажать кнопку «Submit»;</li> <li>5. Перейти на страницу просмотра заказов нажав кнопку «Order List» в меню сайта;</li> <li>6. Нажать кнопку «View» в таблице заказов, чтобы открыть информацию по нему;</li> </ol>	<p>Ожидаемый результат:</p> <ol style="list-style-type: none"> <li>1. После аутентификации на сайте в качестве администратора в меню сайта появятся две кнопки в меню: «Create Product» и «Order List», а также кнопка «Edit Product».</li> <li>2. При нажатии кнопки «Edit Product» открывается окно с редактируемой информацией товара.</li> <li>3. После нажатия кнопки «Submit» в базе данных будет изменена строчка данного товара в таблице «Products».</li> <li>4. В случае неправильного ввода данных либо отправки пустого поля выведется сообщение об ошибке.</li> <li>5. При нажатии кнопки «Create Product» откроется страница для ввода данных нового товара.</li> <li>6. После нажатия кнопки «Submit» в базе данных будет создана строчка в таблице «Products»</li> <li>7. После перехода на страницу заказов по нажатию кнопки «Order List» появится таблица со всеми заказами.</li> </ol>

	8. При нажатии кнопки «View» будет показана информация по выбранному заказу.	
Статус «Разработано»	Дата выполнения 15.05.21	Ссылка на дефекты:

Тест ожидаемого результата №1 изображён на рисунке 3.11.

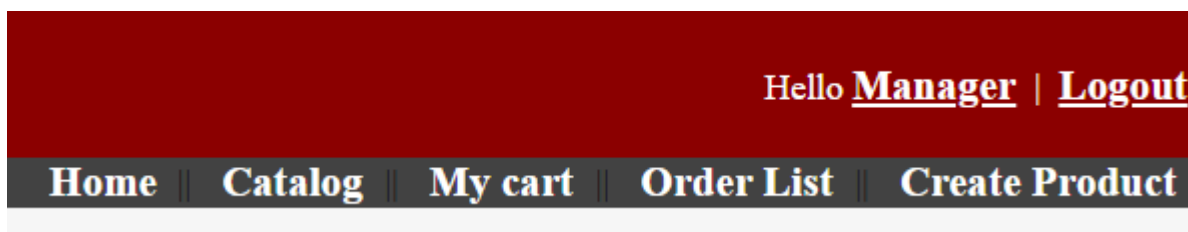


Рисунок 3.11 - Результат теста ожидаемого результата №1

Тест ожидаемого результата №2 изображён на рисунке 3.12.

Рисунок 3.12 - Результат теста ожидаемого результата №2

Тест ожидаемого результата №5 изображён на рисунке 3.13.



The screenshot shows a web form titled 'Estore' with a 'Product' section. It contains input fields for 'Code \*' (S001), 'Name \*' (SSD1), and 'Price \*' (220.0). There is an 'Image' field with a small icon and an 'Upload Image' section with a file selection button labeled 'Выберите файл' and the filename 'SSD1.png'. At the bottom are 'Submit' and 'Reset' buttons.

Рисунок 3.13 - Результат теста ожидаемого результата №5

Тест ожидаемого результата №6 изображён на рисунке 3.14.

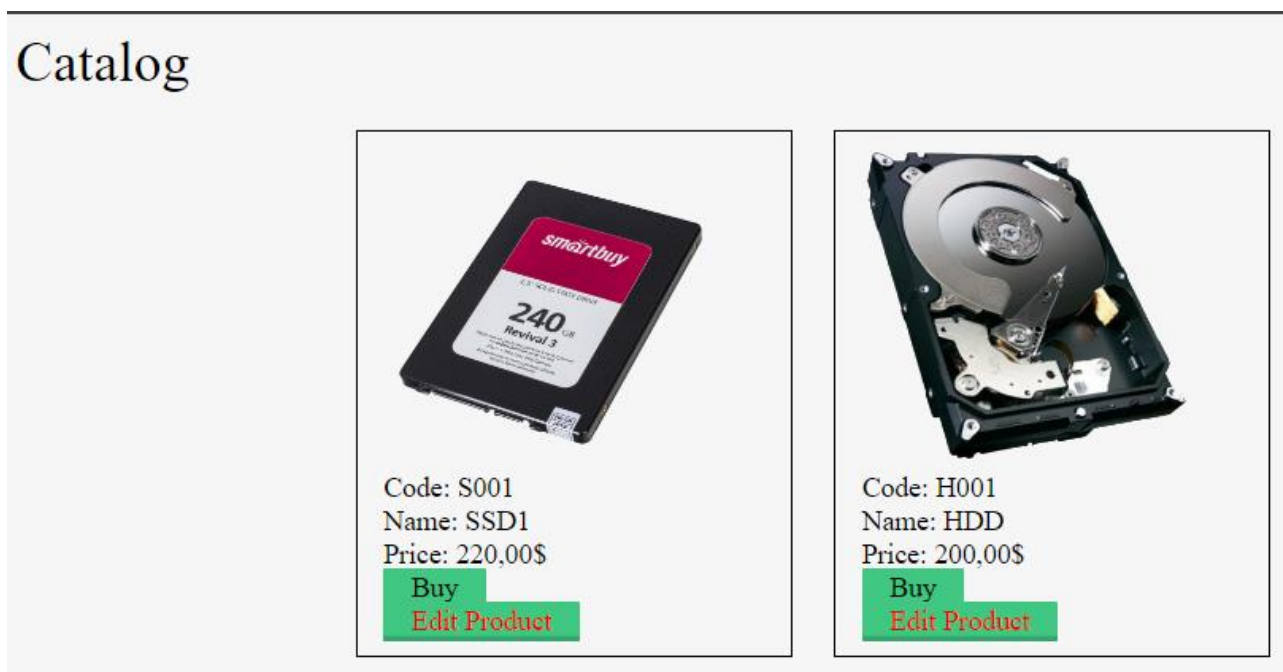


Рисунок 3.14 - Результат теста ожидаемого результата №6



Тест ожидаемого результата №7 изображён на рисунке 3.15.

Total Order Count: 2						
Order Num	Order Date	Customer Name	Customer Address	Customer Email	Amount	View
2	21-05-2021 18:19	Customer2	Slovakia	customer2@mail.ru	2957.00	<a href="#">View</a>
1	11-05-2021 14:43	Neko	Ukraine	customer@mail.ru	2420.00	<a href="#">View</a>

Рисунок 3.15 - Результат теста ожидаемого результата №7

Тест ожидаемого результата №7 изображён на рисунке 3.16.

Product Code	Product Name	Quantity	Price	Amount
R001	RAM	2	110,00	220,00
V001	Video Card	2	1000,00	2000,00
C001	CPU	1	120,00	120,00
H001	HDD	2	200,00	400,00
P001	Power Supply Unit	1	060,00	060,00
M002	Motherboard2	1	157,00	157,00

Рисунок 3.16 - Результат теста ожидаемого результата №7

### Выводы к разделу 3

В данном разделе был составлен тестовый план в соответствии со стандартом «IEEE 829», который позволит повысить отказоустойчивость и стабильность веб-приложения. Были протестированы все функции в соответствии с планом.

## **ЗАКЛЮЧЕНИЕ**

В процессе выполнения данной работе было разработано и протестировано веб-приложение «Магазин компьютерный комплектующих». Спроектированное веб-приложение даёт пользователям возможность просмотра и покупки компьютерных комплектующих, не регистрируясь на сайте. А пользователи с ролью «Администратор», в свою очередь, имеют возможность просмотра заказов, а также изменения и добавления товаров на сайт.

Был проведён анализ предметной области с выявлением базовых функций, которыми должно обладать веб-приложение, технологий, с помощью которых, данное веб-приложение должно быть разработано, а также определены требования, которые должны быть реализованы и протестированы в рамках представленной системы.

Были изучены и применены при разработке необходимые технологии, а также были протестированы сформированные требования, позволяющие облегчить работу с веб-приложением.

Был разработан план тестирования для повышения качества приложения и составлены тест-кейсы в виде демонстрации работы интерфейса и функций приложения.

В процессе разработки работы были приобретены следующие компетенции:

ПК-1 - способен выполнять разработку и интеграцию программных модулей и компонент, верификацию выпусков программных продуктов информационных систем

ПК-1.1 - Знать: методы и средства сборки модулей и компонент программного обеспечения при создании информационных систем

ПК-1.12 - Уметь: применять методы и средства создания программных интерфейсов информационных систем

ПК-1.14 - Владеть: разработкой процедур сборки модулей и компонент программного обеспечения при внедрении информационных систем.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Официальный веб-сайт [paketaninternet.com](https://paketaninternet.com/) [Электронный ресурс]: Режим доступа: <https://paketaninternet.com/> [Дата обращения: 17.05.21];
2. Руководство по Spring [Электронный ресурс]. Режим доступа: <https://proselyte.net/tutorials/spring-tutorial-full-version/> [Дата обращения: 29.03.21];
3. Spring Framework Documentation [Электронный ресурс]. Режим доступа: <https://docs.spring.io/spring-framework/docs/current/reference/html/web.html> [Дата обращения: 29.03.21];
4. Стелтинг С., Маасен О. Применение шаблонов Java. Библиотека профессионала.: Пер. с англ. — М.: Издательский дом "Вильямс", 2002. - 576 с.: ил. — Парал. тит. англ. [Дата обращения: 29.03.21];
5. Functional Interfaces in Java: Fundamentals and Examples 1st ed. Edition, Kindle Edition [Электронный ресурс]. Режим доступа: <https://www.amazon.com/Functional-Interfaces-Java-Fundamentals-Examples-ebook/dp/B07NRHQSCW> [Дата обращения: 29.03.21];
6. Официальный веб-сайт [javarush.ru](https://javarush.ru/) [Электронный ресурс]: Режим доступа: <https://javarush.ru/groups/posts/spring-framework-java-1> [Дата обращения: 17.04.21];
7. Hibernate Search 6.0.0. Final: Reference Documentation [Электронный ресурс]. Режим доступа: [https://docs.jboss.org/hibernate/stable/search/reference/en-US/html\\_single/](https://docs.jboss.org/hibernate/stable/search/reference/en-US/html_single/) [Дата обращения: 29.04.21];
8. Паттерны проектирования на Java. Каталог Java-примеров. [Электронный ресурс]. URL: <https://refactoring.guru/ru/design-patterns/java> [Дата обращения: 09.04.21].
9. The DAO with JPA and Spring [Электронный ресурс]: Режим доступа: <https://www.baeldung.com/spring-dao-jpa> [Дата обращения: 09.04.21].