

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

---

Факультет автоматики и вычислительной техники  
Кафедра вычислительной техники



**НГТУ  
НЭТИ**

## **ОТЧЁТ**

### **КУРСОВОЙ ПРОЕКТ**

по дисциплине: «Программное обеспечение информационных систем»

*Тема: Интернет-магазин телефонов*

Выполнил:  
студент гр. АВТ-010 Казанцев К.О.  
«\_\_» \_\_\_\_\_ 2023 г.

Проверил(а):  
Менжулин С.А.  
«\_\_» \_\_\_\_\_ 2023 г.  
\_\_\_\_\_  
(оценка, подпись)

Новосибирск

2023

## Оглавление

Задание на курсовой проект.....	3
Перечень сокращений и обозначений.....	3
Введение.....	4
1 Предметная область .....	5
1.1 Анализ предметной области .....	5
1.2 Требования к проекту .....	5
2 Проектирование информационной системы .....	6
2.1 Диаграмма классов предметной области.....	6
2.2 Информационные объекты системы.....	8
2.3 Диаграмма состояний .....	12
3 Практическая часть .....	16
3.1 Структура информационной системы .....	16
3.2 Реализация системы.....	16
4 Тестирование .....	19
Заключение .....	26
Использованные источники .....	26

## **Задание на курсовой проект**

Разработать информационную систему на языке C# в соответствии с индивидуальным вариантом задания. Жизненный цикл всех информационных объектов в работе должны быть управляем из интерфейса пользователя или администратора информационной системы.

Индивидуальный вариант задания: Интернет-магазин телефонов.

## **Перечень сокращений и обозначений**

БД – база данных

СУБД – система управления базами данных

ИС – информационная система

DTO – Data Transfer Object

ORM – Object-Relational Mapping

## **Введение**

Современный мир невозможно представить без активного использования мобильных телефонов, которые являются неотъемлемой частью повседневной жизни большинства людей. Развитие технологий и интернета способствовало появлению и распространению интернет-магазинов, предлагающих широкий ассортимент телефонов для самых разных потребностей и бюджетов. Цель данного курсового проекта – разработать и проанализировать эффективную модель интернет-магазина мобильных телефонов, ориентированного на современного потребителя. Проект включает изучение рынка мобильных телефонов, анализ потребностей и предпочтений целевой аудитории, разработку стратегии продвижения и управления ассортиментом, а также создание удобного и функционального интерфейса магазина для обеспечения максимально комфортных условий для выбора и покупки товаров.

## **1 Предметная область**

### **1.1 Анализ предметной области**

Рынок мобильных телефонов характеризуется быстрыми изменениями и доминированием нескольких ведущих брендов. Это требует постоянного отслеживания текущих трендов и адаптации к меняющимся потребностям клиентов. Потребители сегодня ориентированы на инновационные функции, такие как высококачественные камеры и улучшенная автономность устройств, а также ценят удобство использования и доступные цены.

Сфера технологий постоянно развивается, внося инновации, такие как поддержка 5G, складные экраны и повышенная производительность, которые становятся ключевыми факторами выбора для потребителей. Важность онлайн-торговли также растет, особенно в контексте электронной коммерции, где важными факторами являются эффективная логистика, безопасность платежных операций и конкурентоспособное ценообразование.

Разработка интернет-магазина телефонов предлагает множество преимуществ. Это позволяет значительно расширить рынок за счет привлечения клиентов, не ограниченных географически, и увеличить продажи за счет круглосуточной доступности. Также это снижает операционные расходы, устраняя необходимость в физических торговых точках. Важным аспектом является возможность персонализированного маркетинга и более глубокий анализ данных покупателей, что способствует более точной настройке бизнес-стратегии. К тому же, интернет-магазин легко адаптируется к изменениям рынка и позволяет масштабировать бизнес.

### **1.2 Требования к проекту**

Интернет-магазин телефонов должен обеспечивать хранение данных о товарах, имеющимся в ассортименте, брендах и заказах.

Система должна отвечать следующим функциональным требованиям:

- Создание учетных записей пользователей.
- Просмотр всех товаров в ассортименте.

- Просмотр товаров определенной категории (смартфоны, сотовые телефоны).
- Просмотр брендов.
- Просмотр товаров определенного бренда.
- Поиск товаров по названию.
- Просмотр информации о конкретном товаре.
- Добавление товаров в корзину.
- Отображение итоговой стоимости товаров в корзине.
- Оформление заказа.
- Создание, редактирование и удаление товаров и брендов для администраторов.

Требования к графическому интерфейсу:

- Система должна быть реализована в виде многостраничного веб-приложения.
- Все страницы веб-сайта должны быть оформлены в одном стиле.
- Выбор веб-страниц должен осуществляться через навигационное меню.
- Наличие главной страницы с каталогом всех товаров.
- Отдельная страница для администратора.
- Формы для ввода и изменения данных о товаре.
- Формы для ввода и изменения данных о брендах.
- Формы регистрации и авторизации

## **2 Проектирование информационной системы**

### **2.1 Диаграмма классов предметной области**

Ниже приведена диаграмма классов предметной области, описывающая общую структуру иерархии классов системы.

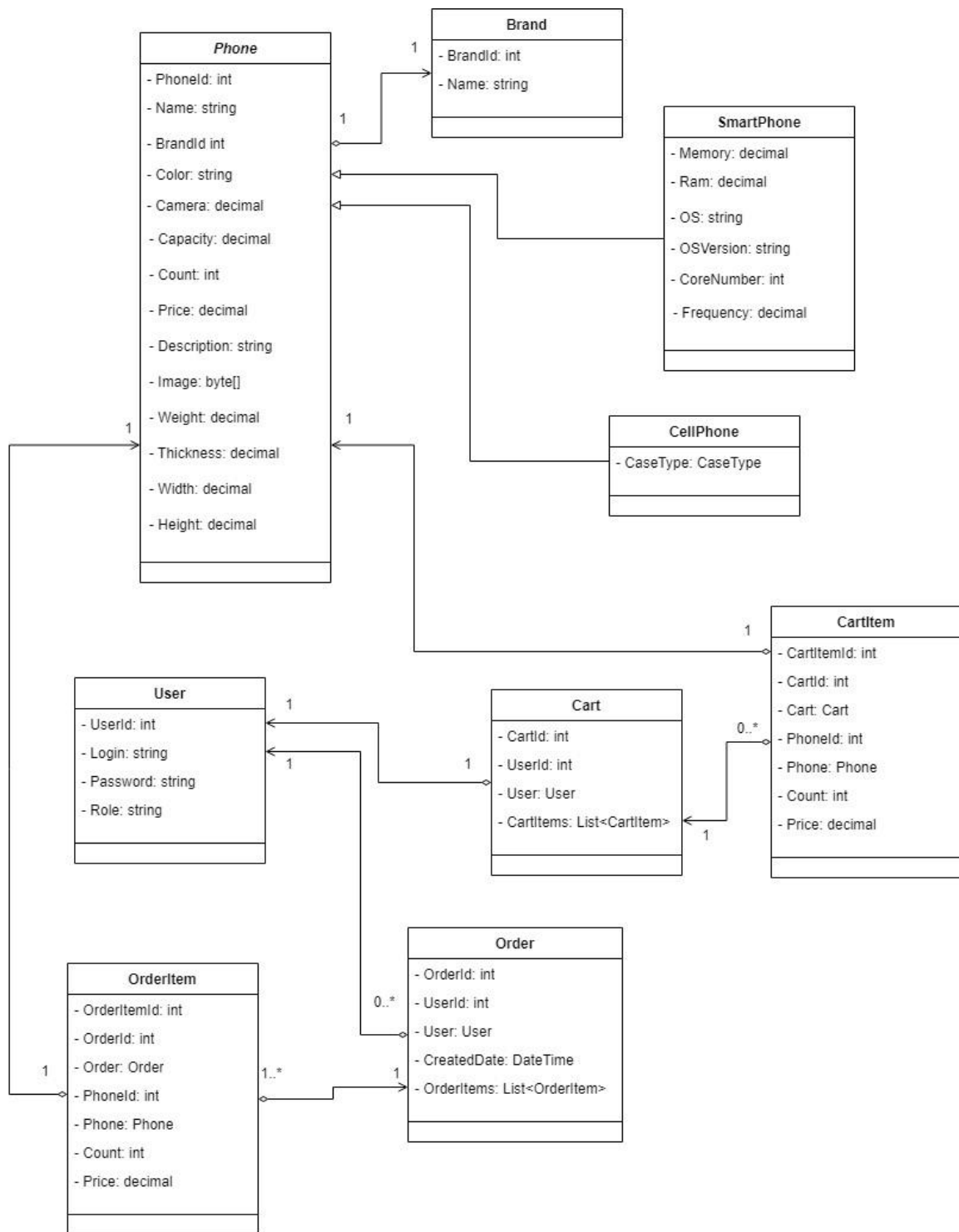


Рисунок 1. Классы предметной области

## 2.2 Информационные объекты системы

К информационным объектам системы относятся таблицы и пользовательские функции, хранимые в базе данных. База данных содержит в себе следующие таблицы:

1. таблица «Phone» - хранит данные об имеющихся товарах;
2. таблица «Brand» - хранит данные об имеющихся брендах;
3. таблица «User» - хранит данные о пользователях;
4. таблица «Cart» - хранит информацию о корзине пользователя;
5. Таблица «CartItem» - хранит данные о товарах в корзине пользователя;
6. таблица «Order» - хранит данные о заказах пользователя;
7. таблица «OrderItem» - хранит данные о товарах в заказах пользователя.

Структура таблиц «Phone», «Brand», «User», «Cart», «CartItem», «Order» и «OrderItem» представлена в таблицах 1, 2, 3, 4, 5, 6 и 7 соответственно.

Таблица 1. Структура таблицы «Phone».

Название поля	Тип поля	Допустимость NULL
PhoneId	integer	not null
Name	text	not null
BrandId	integer	not null
Color	text	not null
Camera	numeric	not null
Capacity	numeric	not null
Height	numeric	not null
Width	numeric	not null
Thickness	numeric	not null
Weigth	numeric	not null
Image	bytea	
Description	text	not null
Price	numeric	not null
Count	integer	not null
PhoneType	character varying(13)	not null
CaseType	integer	
Memory	numeric	



Ram	numeric	
OS	text	
OSVersion	text	
CoreNumber	integer	
Frequency	numeric	
<b>Индексы:</b>		
"PK_Phone" PRIMARY KEY, btree ("PhoneId") "IX_Phone_BrandId" btree ("BrandId")		
<b>Ограничения внешнего ключа:</b>		
"FK_Phone_Brand_BrandId" FOREIGN KEY ("BrandId") REFERENCES "Brand"("BrandId") ON DELETE CASCADE		
<b>Ссылки извне:</b>		
TABLE ""CartItem"" CONSTRAINT "FK_CartItem_Phone_PhoneId" FOREIGN KEY ("PhoneId") REFERENCES "Phone"("PhoneId") ON DELETE CASCADE TABLE ""OrderItem"" CONSTRAINT "FK_OrderItem_Phone_PhoneId" FOREIGN KEY ("PhoneId") REFERENCES "Phone"("PhoneId") ON DELETE CASCADE		

Таблица 2. Структура таблицы «Brand».

Название поля	Тип поля	Допустимость NULL
BrandId	integer	not null
Name	text	not null
<b>Индексы:</b>		
"PK_Brand" PRIMARY KEY, btree ("BrandId")		
<b>Ссылки извне:</b>		
TABLE ""Phone"" CONSTRAINT "FK_Phone_Brand_BrandId" FOREIGN KEY ("BrandId") REFERENCES "Brand"("BrandId") ON DELETE CASCADE		

Таблица 3. Структура таблицы «User».

Название поля	Тип поля	Допустимость NULL
UserId	integer	not null
Login	text	not null
Password	text	not null
Role	text	not null
<b>Индексы:</b>		
"PK_User" PRIMARY KEY, btree ("UserId")		
<b>Ссылки извне:</b>		
TABLE ""Cart"" CONSTRAINT "FK_Cart_User_UserId" FOREIGN KEY ("UserId") REFERENCES "User"("UserId") ON DELETE CASCADE TABLE ""Order"" CONSTRAINT "FK_Order_User_UserId" FOREIGN KEY ("UserId") REFERENCES "User"("UserId") ON DELETE CASCADE		

Таблица 4. Структура таблицы «Cart».

Название поля	Тип поля	Допустимость NULL
CartId	integer	not null
UserId	integer	not null
<b>Индексы:</b>		
"PK_Cart" PRIMARY KEY, btree ("CartId") "IX_Cart_UserId" btree ("UserId")		
<b>Ограничения внешнего ключа:</b>		
"FK_Cart_User_UserId" FOREIGN KEY ("UserId") REFERENCES "User"("UserId") ON DELETE CASCADE		
<b>Ссылки извне:</b>		
TABLE ""CartItem"" CONSTRAINT "FK_CartItem_Cart_CartId" FOREIGN KEY ("CartId") REFERENCES "Cart"("CartId") ON DELETE CASCADE		

Таблица 5. Структура таблицы «CartItem».

Название поля	Тип поля	Допустимость NULL
CartItemid	integer	not null
CartId	integer	not null
PhoneId	integer	not null
Count	integer	not null

Price	numeric	not null
<b>Индексы:</b>		
"PK_CartItem" PRIMARY KEY, btree ("CartItemId") "IX_CartItem_CartId" btree ("CartId") "IX_CartItem_PhoneId" btree ("PhoneId")		
<b>Ограничения внешнего ключа:</b>		
"FK_CartItem_Cart_CartId" FOREIGN KEY ("CartId") REFERENCES "Cart"("CartId") ON DELETE CASCADE "FK_CartItem_Phone_PhoneId" FOREIGN KEY ("PhoneId") REFERENCES "Phone"("PhoneId") ON DELETE CASCADE		

Таблица 6. Структура таблицы «Order».

Название поля	Тип поля	Допустимость NULL
OrderId	integer	not null
UserId	integer	not null
CreatedDate	timestamp with time zone	not null
<b>Индексы:</b>		
"PK_Order" PRIMARY KEY, btree ("OrderId") "IX_Order_UserId" btree ("UserId")		
<b>Ограничения внешнего ключа:</b>		
"FK_Order_User_UserId" FOREIGN KEY ("UserId") REFERENCES "User"("UserId") ON DELETE CASCADE		
<b>Ссылки извне:</b>		
TABLE ""OrderItem"" CONSTRAINT "FK_OrderItem_Order_OrderId" FOREIGN KEY ("OrderId") REFERENCES "Order"("OrderId") ON DELETE CASCADE		

Таблица 7. Структура таблицы «OrderItem».

Название поля	Тип поля	Допустимость NULL
OrderItemId	integer	not null
OrderId	integer	not null
PhoneId	integer	not null
Count	integer	not null
Price	numeric	not null
<b>Индексы:</b>		

"PK_OrderItem" PRIMARY KEY, btree ("OrderItemId") "IX_OrderItem_OrderId" btree ("OrderId") "IX_OrderItem_PhoneId" btree ("PhoneId")
<b>Ограничения внешнего ключа:</b>
"FK_OrderItem_Order_OrderId" FOREIGN KEY ("OrderId") REFERENCES "Order"("OrderId") ON DELETE CASCADE "FK_OrderItem_Phone_PhoneId" FOREIGN KEY ("PhoneId") REFERENCES "Phone"("PhoneId") ON DELETE CASCADE

Связи между таблицами изображены на рисунке 2.

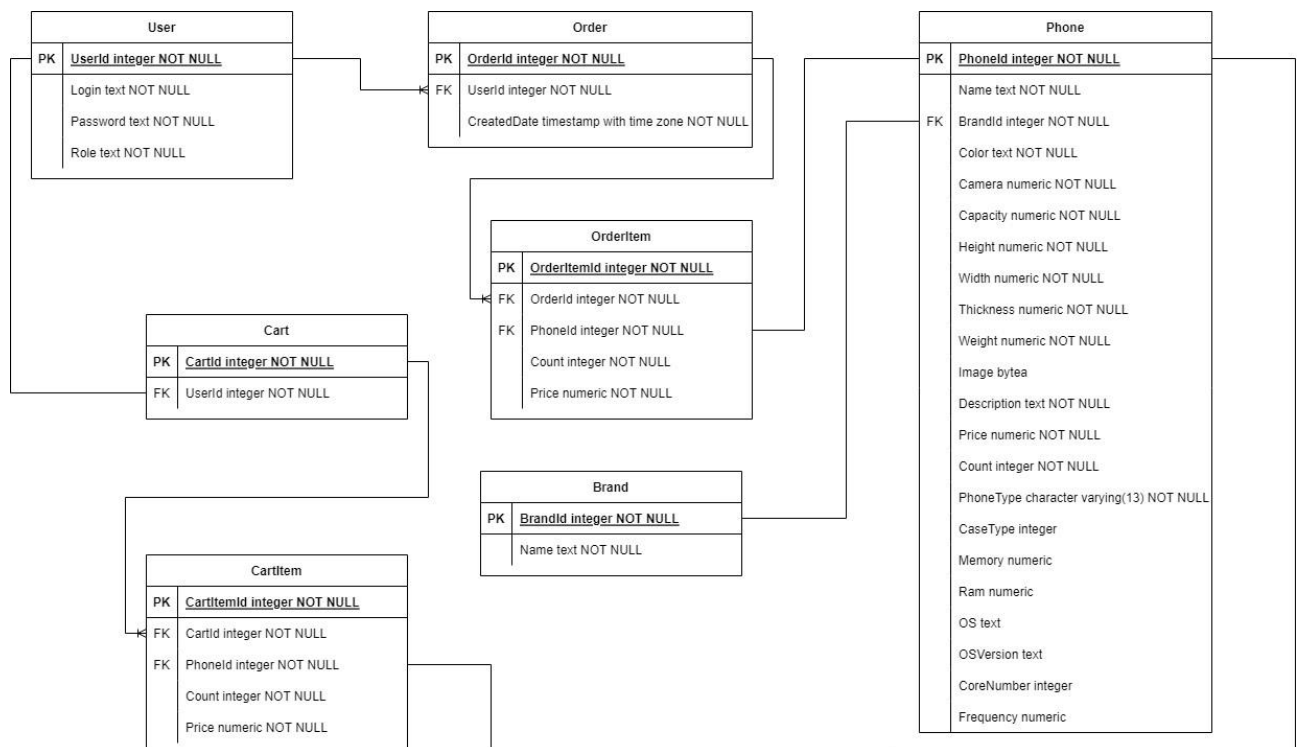


Рисунок 2. Связи между таблицами базы данных

## 2.3 Диаграмма состояний

Диаграммы состояний для процессов регистрации, авторизации и создания заказа приведены на рисунке 3.

При регистрации пользователь вводит необходимые данные в форму. Далее проверяется наличие пользователя в базе данных. Если пользователя с такими данными нет, то он заносится в базу данных и для него создается

собственная корзина в базе данных. После чего происходит авторизация на сайте и переход на главную.

При авторизации пользователь вводит логин и пароль в форму. Далее проверяется наличие пользователя в базе данных. Если пользователь с таким логином обнаружен происходит сопоставление паролей. В случае успешной авторизации производится переход на главную страницу.

При создании заказа пользователь выбирает доступный товар из каталога. В случае, если пользователь авторизован на сайте, он может добавить товар в корзину. Для каждого товара в корзине создается запись в бд. После удаления товара из корзины запись удаляется. После добавления в корзину всех желаемых товаров можно оформить заказ. При оформлении заказа создается новая запись заказа в БД и записи для каждого товара из корзины для данного заказа. Также уменьшается доступное количество товара.

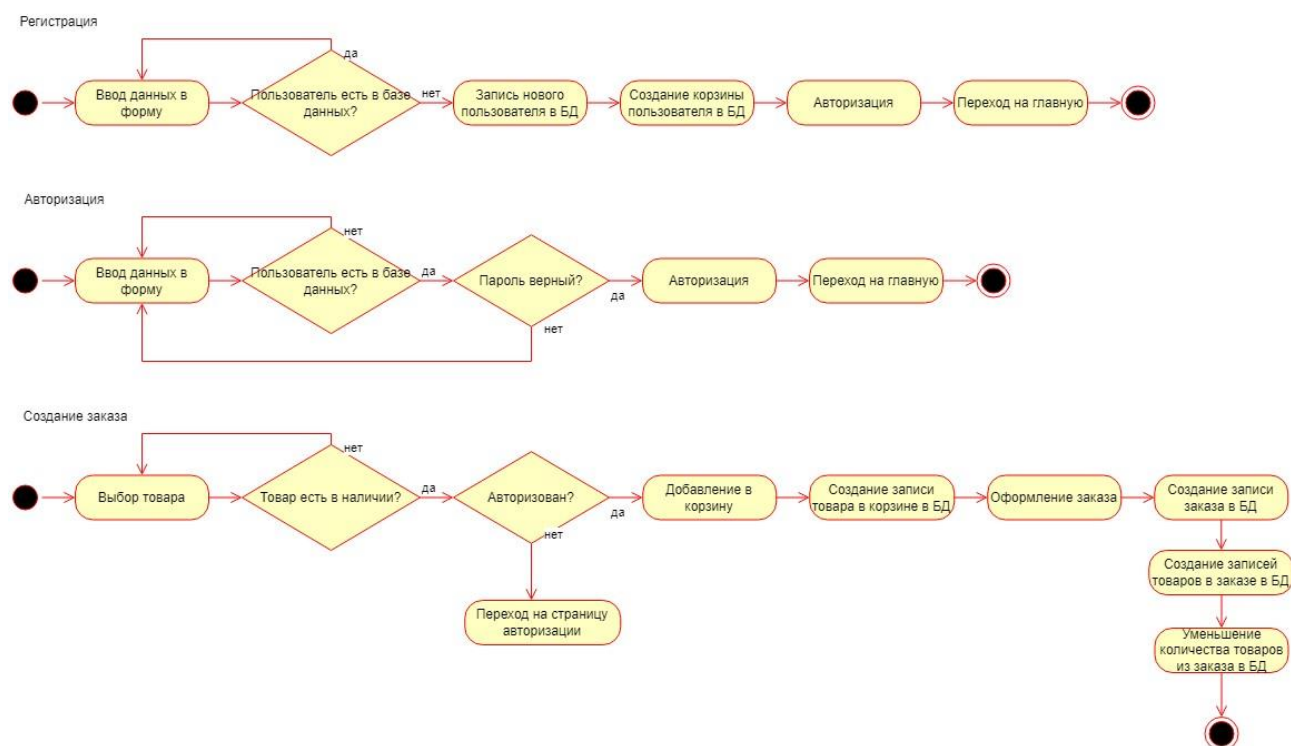


Рисунок 3. Диаграмма состояний

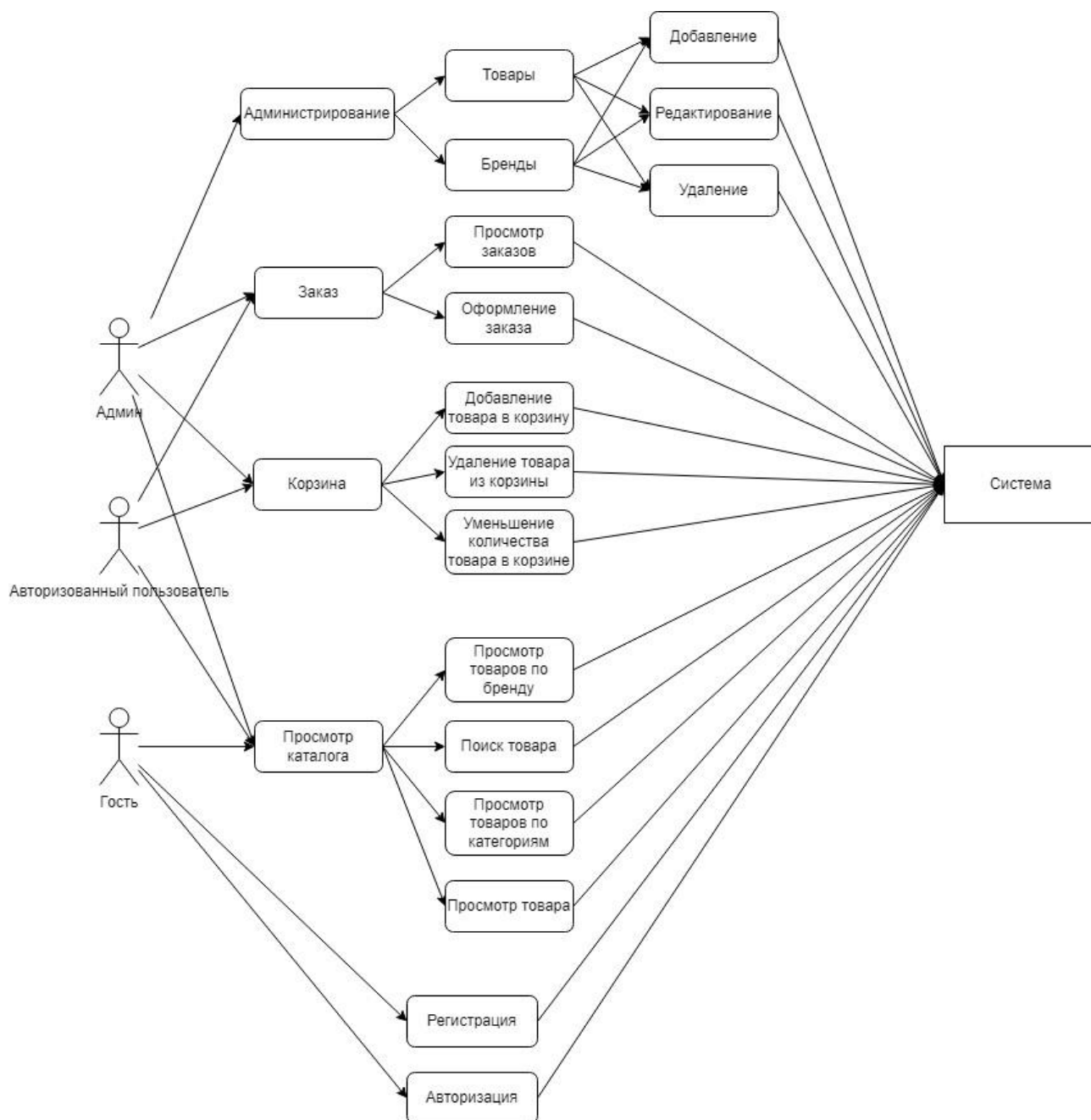


Рисунок 4. Диаграмма прецедентов

Диаграмма прецедентов отображена на рисунке 4. Данная диаграмма содержит 3 основных актора: «Авторизованный пользователь», «Гость», «Админ». Эти актеры взаимодействуют с различными частями системы, которая обозначена как «Система».

«Гость» имеет ограниченный функционал. Он может просматривать товары в каталоге, искать товары по названию, категориям и брендам. А также просматривать страницу товара.

«Авторизованный пользователь» имеет те же функции, что и гость, но вдобавок может добавлять товары в корзину и оформлять заказы.

«Админ» имеет полный доступ к функционалу сайта, включая добавление, редактирование и удаление товаров и брендов.

На рисунке 5 изображена диаграмма компонент реализации системы. Пользовательский интерфейс представлен в виде многостраничного сайта с разделением страниц по функционалу. В качестве языка реализации используется C#. Для пользовательского графического интерфейса используются каскадные таблицы стилей css и фреймворк bootstrap для упрощения стилизации. В качестве СУБД выбран PostgreSQL, так как он бесплатен, осуществляет функции хранения данных в виде реляционных БД и функции манипулирования ими посредством запросов SQL.

Реализация системы онлайн-магазина телефонов включает в себя:

Frontend – Пользовательский интерфейс представлен в виде тонкого клиента (веб-сайт). Взаимодействует с сервером отправкой DTO через формы.

Backend – Серверная часть, состоит из различных сервисов, взаимодействующих с базой данных через репозитории посредством представления таблиц базы данных в виде моделей.

База данных – Хранит данные, необходимые для функционирования магазина.

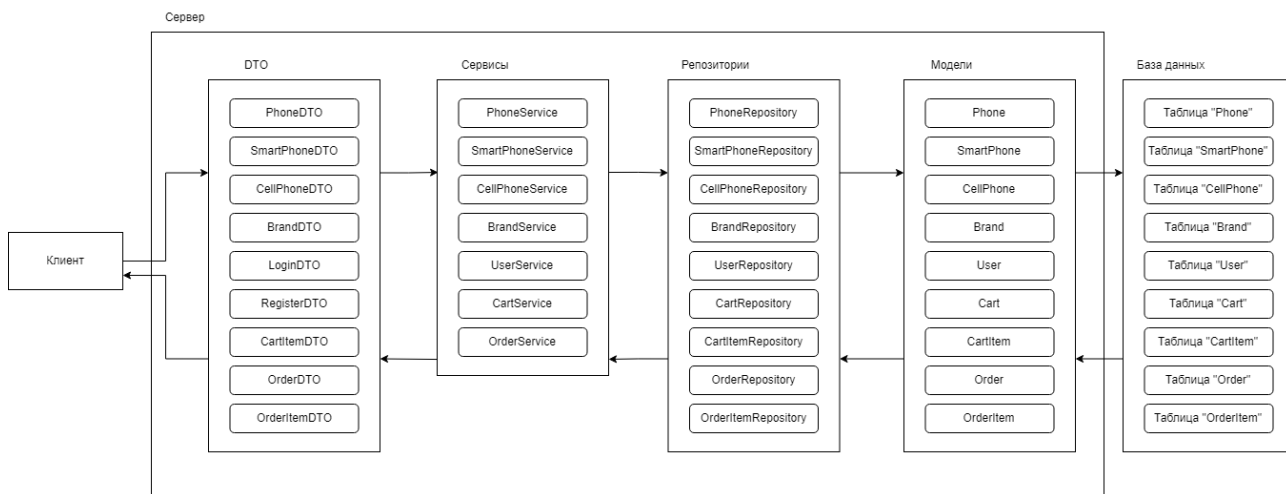


Рисунок 5. Диаграмма компонент реализации системы

## 3 Практическая часть

### 3.1 Структура информационной системы

Для хранения данных интернет-магазина была создана база данных «CourseProject», содержащая следующие таблицы «Phone», «Brand», «User», «Cart», «CartItem», «Order» и «OrderItem».

Заполним некоторые таблицы и выведем их содержимое (рисунки 6-12).

	PhoneId [PK] integer	Name text	BrandId integer	Color text	Camera numeric	Capacity numeric	Height numeric	Width numeric	Thickness numeric	Weight numeric	Image byte	Description text	Price numeric	Count integer	PhoneType character var	CaseType integer	Memory numeric	Ram numeric	OS text	OSVersion text	CoreNumber integer	Frequency numeric
1	1	iPhone 15	1	Черный	48	3349	147.6	71.6	7.8	171	[binary da...	Базовый ...	110999	12	SmartPho...	[null]	128	6	iOS	iOS 17	6	3.46
2	2	iPhone 15	1	Черный	48	3349	147.6	71.6	7.8	171	[binary da...	Базовый ...	125999	0	SmartPho...	[null]	256	6	iOS	iOS 17	6	3.46
3	3	iPhone 15	1	Синий	48	3349	147.6	71.6	7.8	171	[binary da...	Базовый ...	125999	3	SmartPho...	[null]	256	6	iOS	iOS 17	6	3.46
4	4	iPhone 15	1	Синий	48	3349	147.6	71.6	7.8	171	[binary da...	Базовый ...	110999	23	SmartPho...	[null]	128	6	iOS	iOS 17	6	3.46
5	5	C185	2	Черный	0	600	111.2	48.2	13.5	65	[binary da...	Сотовый ...	850	33	CellPhone	0	[null]	[null]	[null]	[null]	[null]	[null]
6	6	V244	2	Черный	0.3	800	100.5	53	18.3	91.7	[binary da...	Сотовый ...	1999	22	CellPhone	1	[null]	[null]	[null]	[null]	[null]	[null]

Рисунок 6. Содержимое таблицы "Phone"

	BrandId [PK] integer	Name text
1	1	Apple
2	2	Dexp

Рисунок 7. Содержимое таблицы "Brand"

	UserId [PK] integer	Login text	Password text	Role text
1	1	Admin	a320480f534776bddb5cdb54b1e93d210a3c7d199e80a23c1b2178497b184c...	Admin

Рисунок 8. Содержимое таблицы "User"

	CartId [PK] integer	UserId integer
1	1	1

Рисунок 9. Содержимое таблицы "Cart"

### 3.2 Реализация системы

Разработка системы осуществляется в среде разработки Visual Studio 2022 Community. В качестве шаблона для реализации многостраничного веб-сайта использован «Веб-приложение ASP.NET Core». При создании приложения был использован подход Razor Pages.



Razor Pages — это подход к созданию веб-приложений в ASP.NET Core, который упрощает программирование страниц с использованием мощных возможностей серверного кода.

Каждая веб-страница представлена отдельным файлом Razor (с расширением `.cshtml`), который содержит как разметку HTML, так и серверный код C#. Это позволяет централизовать обработку на уровне страницы, делая код более понятным и удобным для управления. Каждая страница Razor сопровождается файлом модели страницы (с расширением `.cs`), который содержит логику обработчиков событий и доступ к данным.

Для создания таблиц базы данных и доступа к ней был использован Entity Framework Core. EF Core позволяет разработчикам работать с базами данных, используя объекты .NET, что снижает количество кода, необходимого для доступа к данным.

Для удобной работы с моделями данных были разработаны репозитории моделей, обеспечивающие простейшие операции над ними (создание, обновление, получение, удаление). Данные репозитории в дальнейшем использовались в сервисах, обеспечивающих основную бизнес логику.

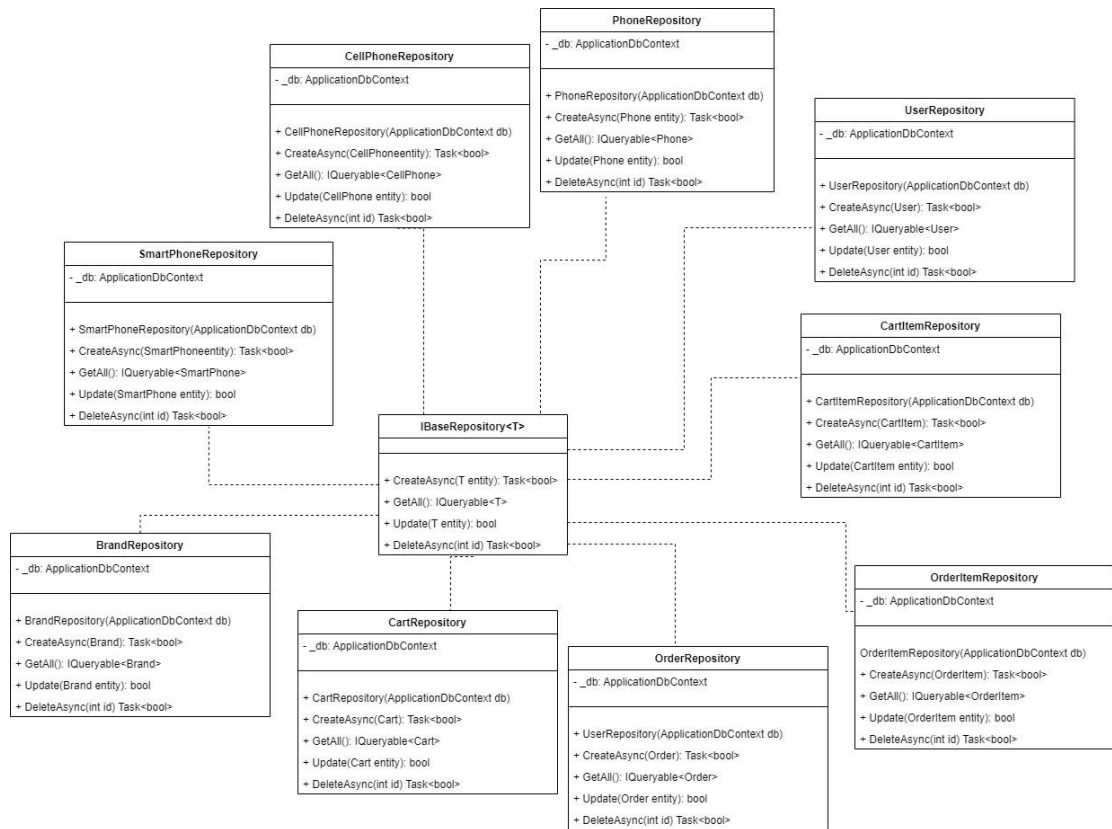


Рисунок 10. Репозитории проекта

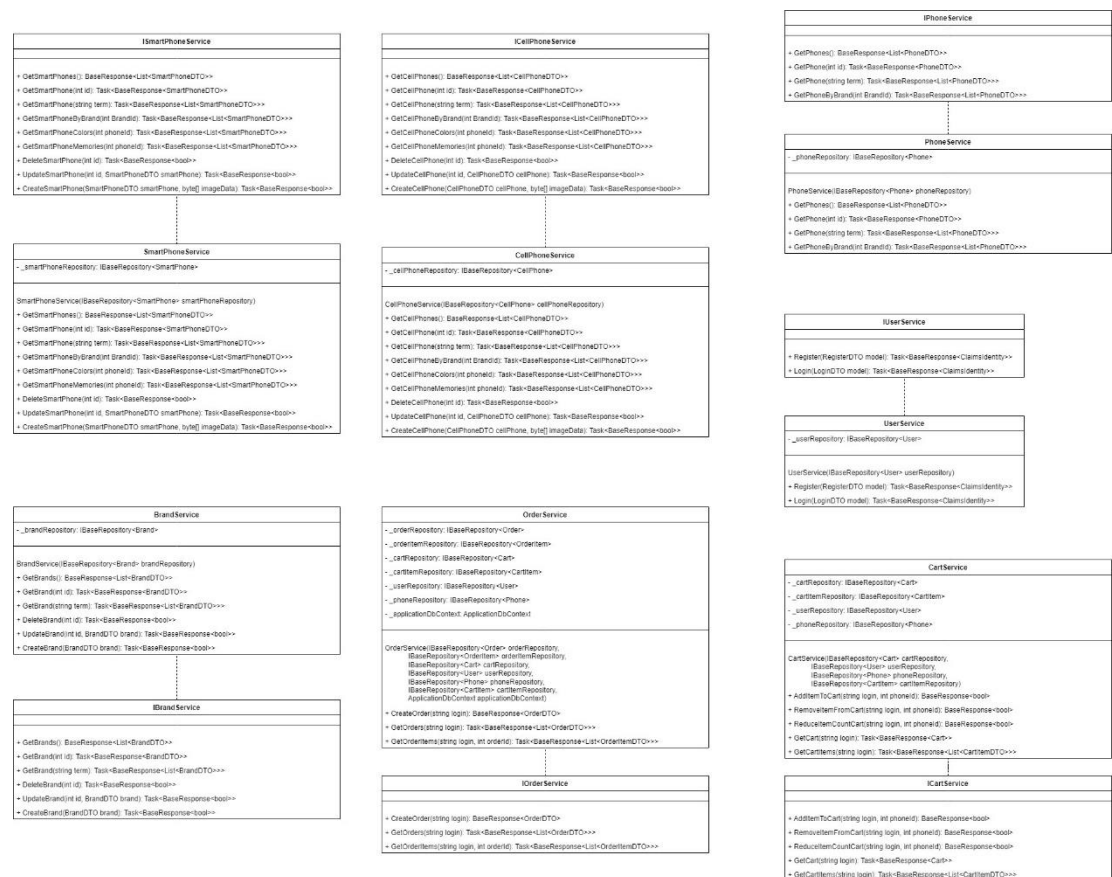


Рисунок 11. Сервисы проекта

## 4 Тестирование

Запустим проект и протестируем приложение «интернет-магазин телефонов».

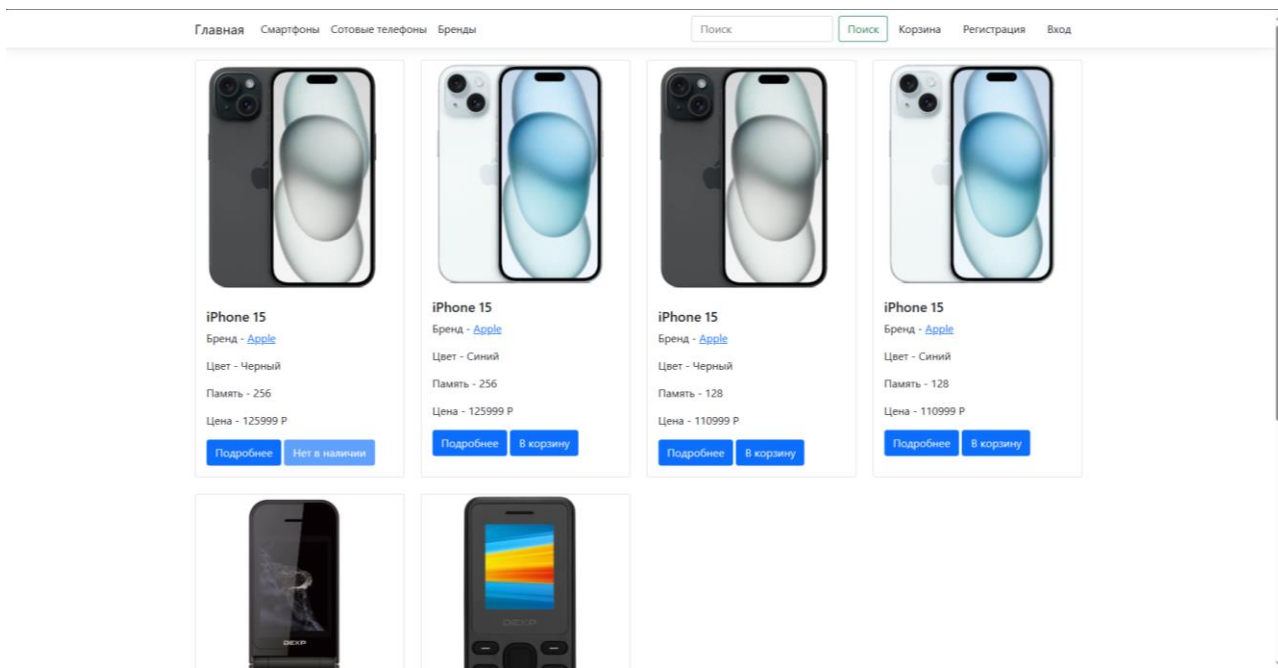


Рисунок 12. Главная страница

На главной странице представлены все товары в каталоге, включая отсутствующие в продаже в данный момент.

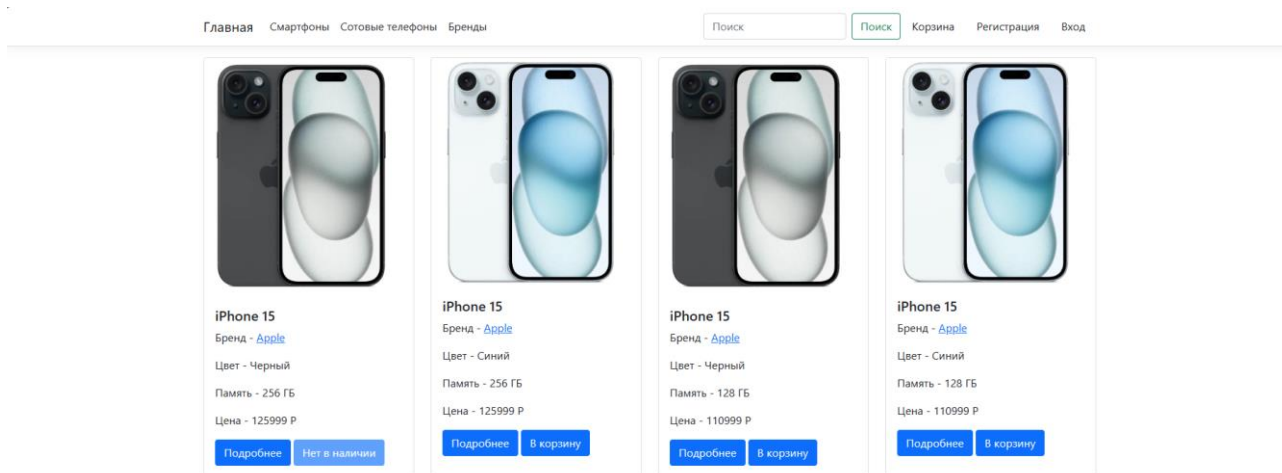


Рисунок 13. Страница смартфонов

На странице смартфонов представлены только смартфоны. А на странице Сотовые телефоны только сотовые соответственно.

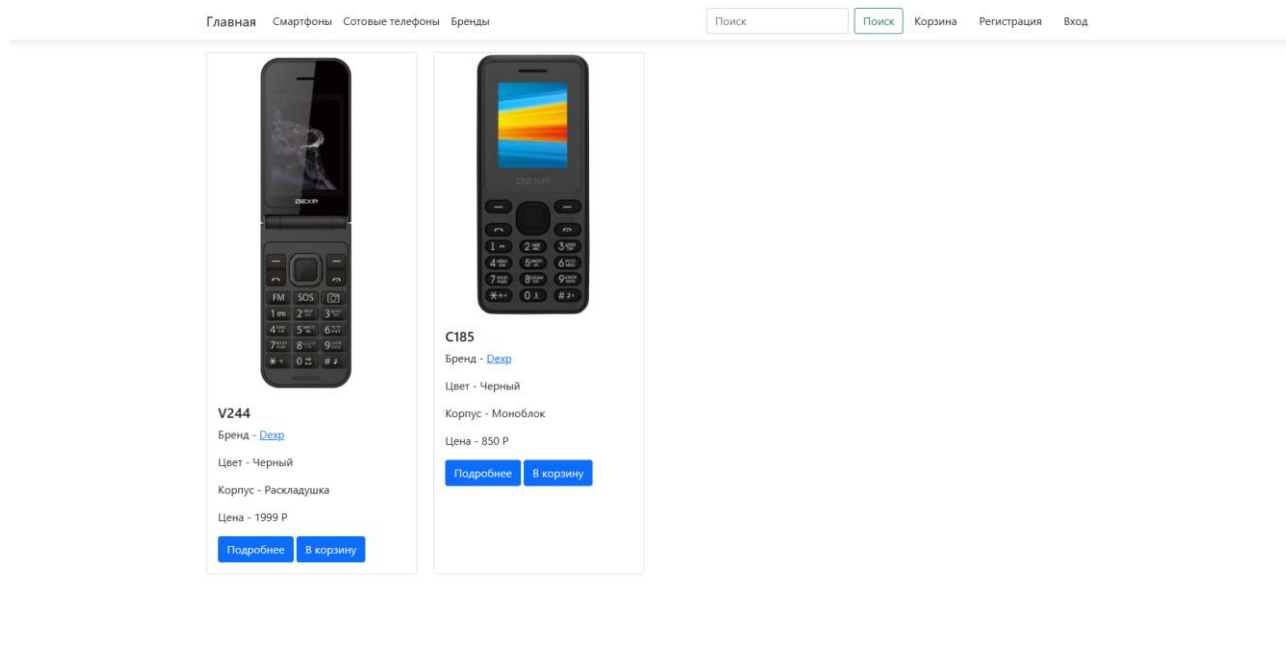


Рисунок 14. Сотовые телефоны

Имеется поиск по названию.

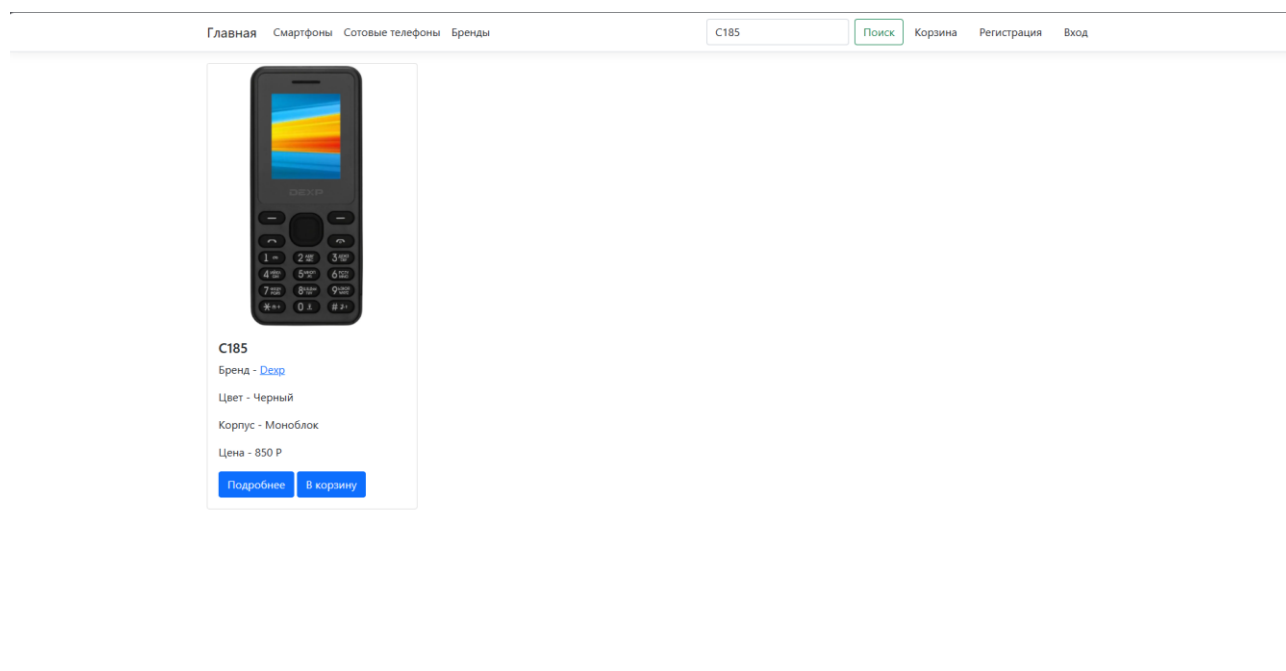


Рисунок 15. Поиск товара по названию

На странице брендов указаны, имеющиеся на сайте бренды, с возможностью просмотреть товары определенного бренда.



Рисунок 16. Страница брендов

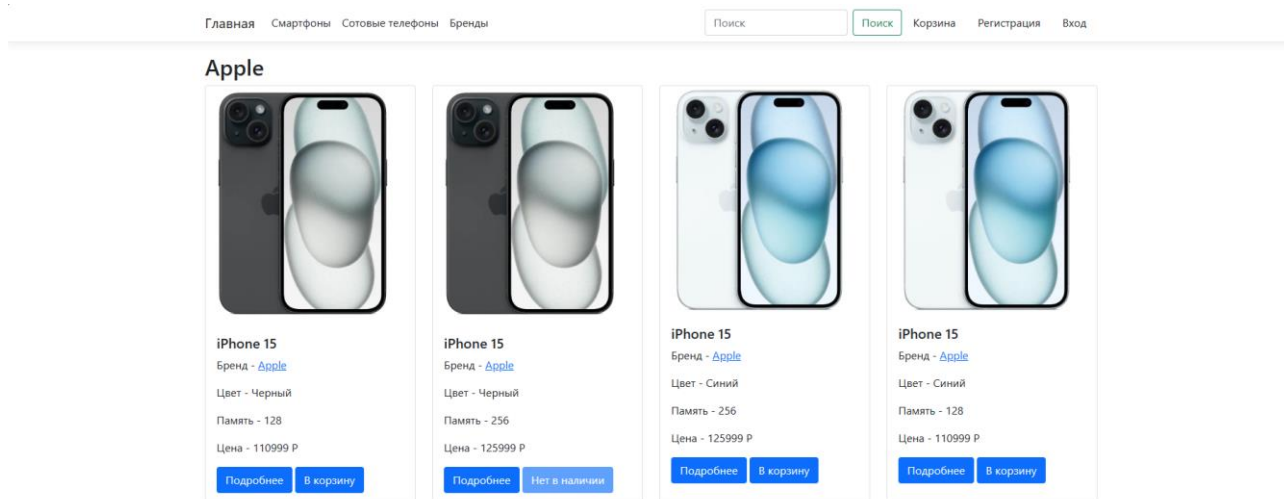


Рисунок 17. Фильтрация по бренду Apple

При нажатии кнопки «Подробнее» на карточке товара мы попадаем на страницу товара. Для смартфонов представлена возможность выбора модели с определенным количеством внутренней памяти и определенного цвета, если таковые имеются. Для сотовых телефонов возможен выбор только по цвету.

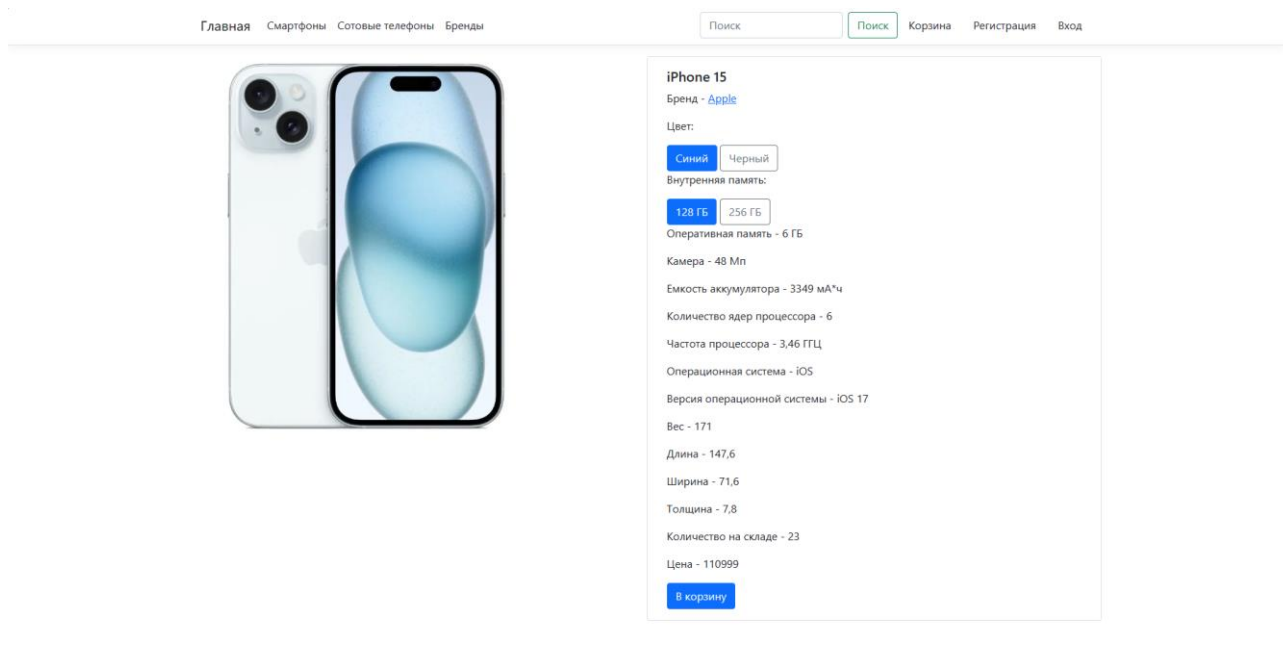


Рисунок 18. Страница товара

При добавлении товара в корзину попадаем на страницу авторизации

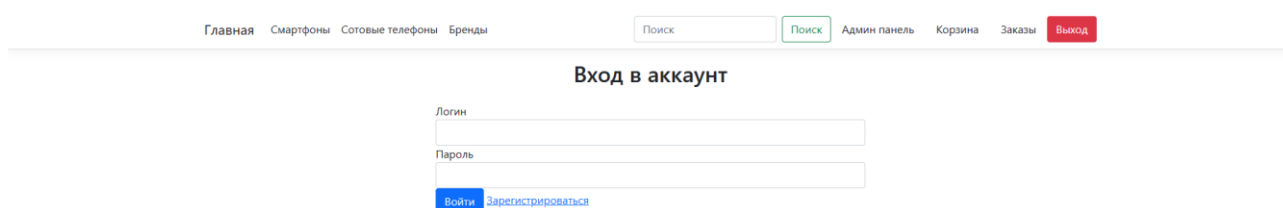


Рисунок 19. Страница авторизации

Добавим товар в корзину и перейдем на страницу корзины. Здесь можно изменить количество товара или убрать ненужный. Также можно узнать итоговую стоимость товаров и оформить заказ.

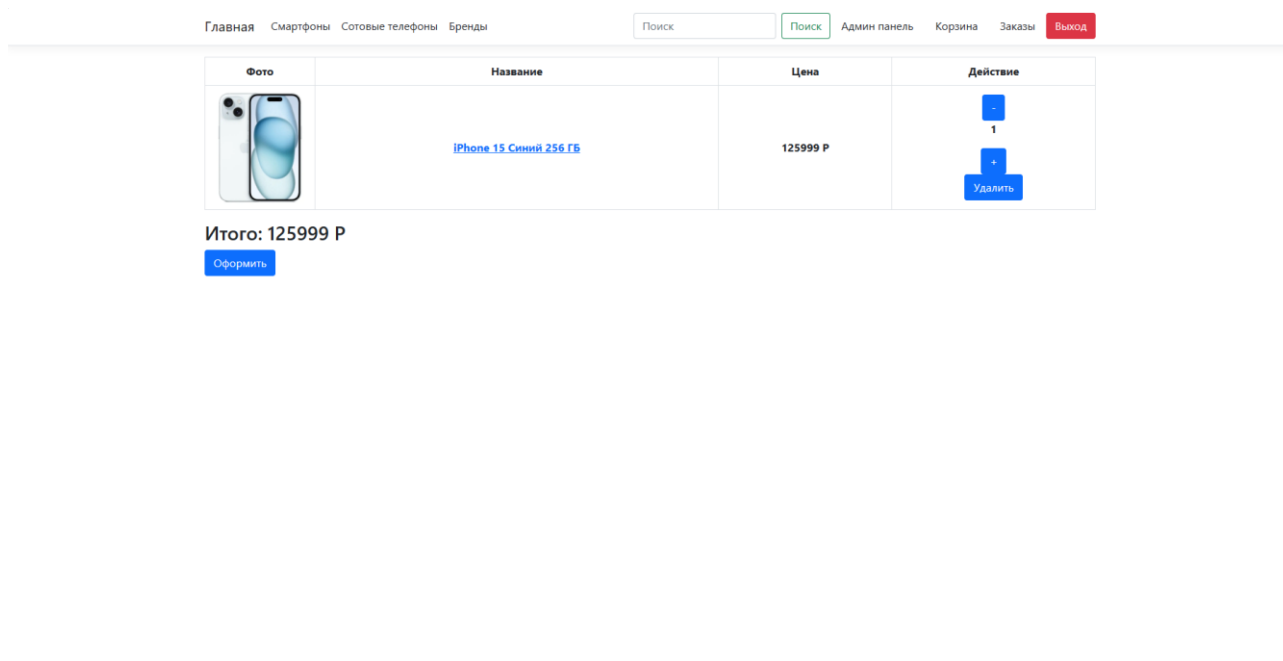


Рисунок 20. Страница корзины

При оформлении заказа попадаем на страницу заказов, где можно посмотреть информацию о предыдущих заказах.

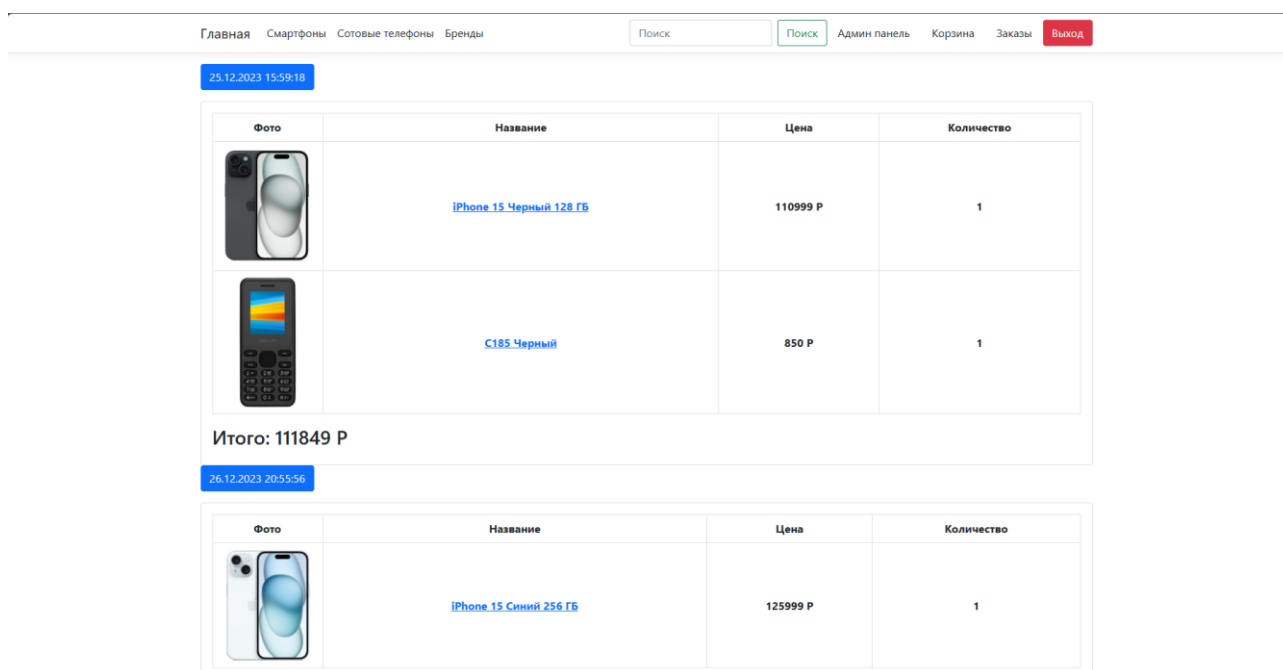


Рисунок 21. Страница заказов

У администратора системы есть доступ к админ панели.

Рисунок 22. Админ панель

Перейдем на страницу администрирования сотовыми телефонами. Здесь нас ждет список, имеющих телефонов с возможностью удалить или изменить их. Также есть кнопка добавить новый телефон.

[Главная](#)
[Смартфоны](#)
[Сотовые телефоны](#)
[Бренды](#)

[Админ панель](#)
[Корзина](#)
[Заказы](#)

[Назад](#)  
[Добавить телефон](#)



Фото	Название	Тип корпуса	Цвет	Действие
	V244	Раскладушка	Черный	<input type="button" value="Изменить"/> <input type="button" value="Удалить"/>
	C185	Моноблок	Черный	<input type="button" value="Изменить"/> <input type="button" value="Удалить"/>

Рисунок 23. Список сотовых телефонов

Перейдем на страницу редактирования сотового телефона. Страница создания телефона идентична.



[Главная](#) [Смартфоны](#) [Сотовые телефоны](#) [Бренды](#)

[Админ панель](#) [Корзина](#) [Заказы](#)

[Назад](#)  
Название  
  
Бренд  
  
Цвет  
  
Тип корпуса  
  
Емкость аккумулятора мА\*ч  
  
Разрешение камеры Мп  
  
Вес г  
  
Длина мм

Ширина мм  
  
Толщина мм  
  
Изображение  
  
Описание товара  

Сотовый телефон DEXP V244 – модель с раскладной конструкцией, созданная для удобства мобильного общения с использованием услуг нескольких операторов связи. Он выполнен в компактном корпусе из прочного пластика и поддерживает установку двух карт SIM. На основном дисплее диагональю 2.4 дюйма TN передается четкая картинка с яркими и красочными цветами. Дополнительный дисплей 1.44 дюйма отображает уведомления и время без необходимости раскрытия корпуса. В телефоне DEXP V244 имеются модуль Bluetooth, проигрыватель MP3, видеоплеер, диктофон, FM-радио. На обратной

  
Цена р  
  
Количество

Рисунок 24. Редактирование телефона

25

## **Заключение**

В процессе реализации данной курсовой работы был создан веб-магазин телефонов на основе C# с применением технологий ASP.NET Core Razor Pages и Entity Framework. Исследование предметной области позволило сформулировать спецификации и требования к проекту. Также было выполнено подробное проектирование информационной системы, включая разработку комплексных диаграмм классов, информационных объектов системы, а также диаграмм состояний, прецедентов и компонентов системы.

Система обеспечивает пользователям широкий спектр функциональных возможностей, начиная от просмотра и выбора товаров до формирования и просмотра заказов. Административный интерфейс позволяет управлять каталогом товаров, категориями и брендами, обеспечивая гибкость и удобство управления контентом магазина.

## **Использованные источники**

1. Документация по ASP.NET [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/aspnet/core/>. Дата обращения: [19.12.2023].
2. Интернет-магазин [Электронный ресурс]: Википедия. Свободная энциклопедия. – Режим доступа: <https://ru.wikipedia.org/wiki/Интернет-магазин> (дата обращения: 10.12.2023).
3. Введение в Razor Pages в ASP.NET Core [Электронный ресурс]: Информационный ресурс Microsoft Learn. – Режим доступа: <https://learn.microsoft.com/ru-ru/aspnet/core/razor-pages/?view=aspnetcore-8.0&tabs=visual-studio> (дата обращения: 20.12.2023).
4. Центр документации Entity Framework [Электронный ресурс]: Информационный ресурс Microsoft Learn. – Режим доступа: <https://learn.microsoft.com/ru-ru/ef/> (дата обращения: 21.12.2023).
5. Руководство по Entity Framework Core 8 [Электронный ресурс]: Информационный ресурс METANIT.COM. – Режим доступа: <https://metanit.com/sharp/efcore/> (дата обращения: 21.12.2023).
6. Руководство по CSS [Электронный ресурс]: Информационный ресурс MDN Web Docs. – Режим доступа: <https://developer.mozilla.org/ru/docs/Web/CSS/Reference> (дата обращения: 24.12.2023).