

Министерство образования и молодежной политики Свердловской области
государственное автономное профессиональное
образовательное учреждение Свердловской области
«Уральский радиотехнический колледж им. А.С. Попова»

Пояснительная записка к курсовому проекту
по МДК 02.02 «Технология разработки и защиты баз данных»
Проектирование и разработка программного обеспечения для магазина
здорового питания «НатурПродукт»
РК 09.02.03.432 06 ПЗ

Нормоконтролер

_____ А. В. Беяева

« ____ » _____ 2023 г.

Руководитель

_____ А. В. Беяева

« ____ » _____ 2023 г.

Разработчик

_____ А. А. Голосов

« ____ » _____ 2023 г.

УТВЕРЖДЕНО

цикловой методической комиссией
информационных технологий
от «___» _____ 20__ г.

«___» _____ Н.Н Шутова

ЗАДАНИЕ

для курсового проектирования по ПМ.03 Участие в интеграции программных модулей по МДК 03.01 Технология разработки программного обеспечения студенту(ке) «4» курса группы «_____»

ГАПОУ СО «Уральский радиотехнический колледж им А.С. Попова»

Тема задания _____

Курсовой проект на указанную тему выполняется студентом в следующем объеме:

1) Пояснительная записка состоит из следующих разделов:

Введение

1. Постановка задачи

2. Системный проект

2.1. Описание предметной области

2.2. Диаграммы вариантов использования

2.3. Сценарии вариантов использования

2.4. Описание данных

2.5. Логическая структура базы данных3 Технический проект

2.6. Выбор состава технических и программных средств

2.7. Физическая структура базы данных

2.8. Физическая структура программы

3. Экспериментальная часть

3.1. Тестирование приложения

3.2. Руководство пользователя.

Заключение

Список использованных источников

Дата выдачи «___» _____ 20__ г

Срок окончания «___» _____ 20__ г

Преподаватель _____ А.В. Беляева

Содержание

Введение.....	4
1 Постановка задачи	6
2 Системный проект	9
2.1 Описание предметной области.....	9
2.2 Диаграммы вариантов использования.....	9
2.3 Сценарии вариантов использования.....	11
2.4 Описание данных.....	16
2.5 Логическая структура базы данных.....	21
3 Технические средства.....	24
3.1 Выбор состава технических и программных средств.....	24
3.2 Физическая структура базы данных	26
3.3 Физическая структура программы.....	28
4 Экспериментальная часть.....	32
4.1 Тестирование приложения.....	32
4.2 Руководство пользователя	36
Заключение	47
Список использованных источников	48

					ПК 09.02.03 423 06 ПЗ			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Голосов А. А.			Проектирование и разработка программного обеспечения для магазина здорового питания "НатурПродукт"	Лит.	Лист	Листов
Провер.		Беляева А. В.					3	
Н. Контр.		Беляева А. В.				ГАПОУ СО УРТК им. А.С. Попова		

Введение

Основанием для проектирования и разработки программного обеспечения для интернет-магазина здорового питания «НатурПродукт» является необходимость упрощения работы с покупателями, и заказами, которые составляют покупатели самостоятельно с помощью программного обеспечения. Магазин здорового питания «НатурПродукт» занимается продажами наборов продуктов здорового питания и оформлением заказов с доставкой на дом.

Торговые организации обязаны проводить свою деятельность в соответствии с множеством требований, в современных условиях торговли. Существуют технологии для автоматизации множества процессов внутри предприятия, например, учёт продаж или оформление заказа.

В нынешнее время стали актуальны услуги заказа товаров с доставкой на дом. Это является простым способом покупки и получения каких-либо товаров, и стало одним из важных требований покупателей к компаниям, занимающихся продажами товаров во множестве сфер.

Данный программный продукт удобен для ведения учёта заказов, доставок и сборе статистики предприятия. В сравнении с аналогичными программными продуктами, данное программное обеспечение не требует каких-либо особых навыков у пользователя, интерфейс и функционал прост и понятен.

Использование данного приложения позволит ускорить работу, автоматизировав большинство действий, которые ранее приходилось делать вручную, что приводило к некорректным и неактуальным данным. С использованием данного программного продукта будет упрощена работа с учётом заказов, афиширования продукции, ведением продаж и статистики магазина.

					ПК 09.02.03 432 06 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

В пояснительной записке представлены четыре раздела:

- постановка задачи. В этом разделе определена цель курсового проекта и поставлены задачи для достижения данной цели;
- системный проект. В этом разделе описана структура работы программного обеспечения и логическая структура базы данных;
- технический проект. В этом разделе выбран состав технических и программных средств для программного обеспечения и описана физическая структура базы данных.
- экспериментальная часть. В данном разделе описано тестирование созданного программного обеспечения, и руководство пользователя.

					РК 09.02.03 432 06 ПЗ	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

1 Постановка задачи

В данной курсовой работе целью является разработка программного обеспечения для автоматизации процессов внутри предприятия «НатурПродукт», для удобного управления деятельностью магазина, его продажами, и сбором статистики.

Для достижения данной цели поставлены задачи:

- описать предметную область. Изучить сферу деятельности магазинов, для корректного понимания процессов внутри предприятия, и понимания какие процессы возможно автоматизировать;
- разработать диаграмму вариантов использования. Изучить и проанализировать данные отражающие роли и задачи предметной области.
- разработать сценарии вариантов использования. Изучить и описать сценарии вариантов использования программного обеспечения.
- описать данные. Изучить какие данные существуют внутри процессов;
- описать логическую структуру базы данных. Разобраться в том, какие образом в базе данных будет происходить обмен информации с сопутствующим ей приложением, как будут храниться данные;
- выбрать состав технических и программных средств. Рассмотреть варианты технических и программных средств, проанализировать какие технологии будут актуальными и наиболее удобными для создания технического проекта.
- создать физическую структуру базы данных. На основе данных и диаграмм, рассмотренных ранее, нужно создать физическую структуры базы данных.
- создать физическую структуры программы. На основе структуры базы данных, необходимо создать приложение по управлению базой данных.

– провести тестирование приложения. После создания физической структуры базы данных и физической структуры приложения, необходимо провести тестирование данных систем на выявление ошибок, которые в последствии необходимо исправить.

– создать руководства пользователя. После тестирования приложения и исправления ошибок, необходимо создать руководство пользователя, которое поможет изучить функционал программного обеспечения.

В программном обеспечении присутствуют 4 вида пользователей:

- гость (неавторизованный пользователь);
- покупатель;
- менеджер заказов;
- администратор базы данных.

Функции гостя:

- просмотр каталога товаров;
- поиск товаров по фильтрации;
- добавление товаров в корзину;
- удаление товаров из корзины;
- регистрация на сайте.

функции покупателя:

- просмотр каталога товаров;
- поиск товаров по фильтрации;
- вход в аккаунт;
- выход из аккаунта;
- добавление товаров в корзину;
- удаление товаров из корзины;
- обращение в поддержку;
- оформление заказа;
- оплата заказа;

- просмотр истории заказов.

Функции администратора базы данных:

- добавление товаров в каталог;
- добавление новых сотрудников;
- редактирование продукции в каталоге;
- отправление ответов на запросы покупателей;
- вход в аккаунт;
- выход из аккаунта.

Функции менеджера заказов:

- отправление ответов на запросы покупателей;
- изменение статуса заказа;
- вход в аккаунт;
- выход из аккаунта.

2 Системный проект

2.1 Описание предметной области

Предприятие «НатурПродукт» занимается продажей различной продукции здорового питания. Для проведения продаж, покупателю необходимо выбрать продукцию, которая его интересует, добавить в корзину выбрать вариант оплаты, выбрать адрес доставки, подтвердить, что информация о доставке является верной, и дожидаться самой доставки, в это время программное обеспечение уведомит администратора сайта о новом заказе, и отправит запрос на формирование и отправку заказа покупателю.

Для корректного отображения продукции, которую возможно заказать, администратору необходимо проводить запросы в цех по формированию заказов, о наличии той или иной продукции. В отсутствие какой-либо продукции, администратор должен указать, что продукция временно недоступна. Данные действия не допускают возможного недопонимания со стороны покупателей и заказов, которые невозможно выполнить.

При подаче заявки на заказ, цех по формированию заказов собирает набор из указанной в заказе продукции. Затем, уже готовый к доставке заказ, передаётся курьеру. У курьера должен быть сам заказ и адрес доставки, который был получен от администратора.

При проведении продаж, доставки заказов и проведении других бизнес-процессов должны формироваться данные о проведении каких-либо операций, в которых находятся данные о заказе (продукты в заказе, их количество, стоимость, номер заказа) и о клиенте (имя, номер телефона, адрес доставки).

2.2 Диаграммы вариантов использования

Диаграммы вариантов использования — это визуальные модели, которые помогают описать, как различные пользователи взаимодействуют с системой, выполняя определенные действия. Для магазина здорового питания

«НатурПродукт» можно создать диаграмму вариантов использования. Данная диаграмма изображена на рисунке 2.2.1:



Рисунок 2.2.1 – Диаграмма вариантов использования

2.3 Сценарии вариантов использования

Для магазина здорового питания могут существовать различные сценарии вариантов использования сайта:

2.3.1 оформление заказа – покупатель просматривает каталог товаров, добавляет нужные в корзину, указывает свои данные для заказа и оформляет покупку.

Основной поток событий:

- система формирует страницу каталога товаров;
- покупатель просматривает продукты в каталоге и добавляет выбранные продукты в корзину;
- система отображает выбранные товары в корзине;
- покупатель указывает необходимые данные для оформления и нажимает кнопку «оформить заказ»;
- система создаёт заказ в базе данных со статусом «в обработке».

Альтернативные потоки:

- покупатель неверно заполнил форму заказа, при неверном заполнении, система выводит сообщение о необходимости исправления тех или иных ошибок в форме.

Предусловия:

- для оформления заказа пользователь должен быть авторизован.

2.3.2 просмотр истории заказов – покупатель просматривает историю заказов на странице «история заказов».

Основные потоки:

- покупатель нажимает на ссылку истории заказов в поле навигации;
- система формирует страницу истории заказов пользователя с информацией о каждом заказе пользователя и его статусом.

Альтернативные потоки:

- Если у пользователя ещё не было ни одного заказа, то система отобразит надпись «ваша история заказов пуста».

2.3.3 добавление товаров в каталог – администратор базы данных, добавляет товары в базу данных.

Основной поток:

- администратор переходит в панель администратора;
- система формирует панель администратора;
- администратор нажимает на кнопку «добавить» у таблицы продуктов;
- система формирует страницу добавления новых товаров;
- администратор добавляет заполняет форму добавления товара и нажимает «добавить»;
- система регистрирует изменения в товарах, обновляя информацию в базе данных и переносит администратора на панель администратора.

Альтернативный поток:

- администратор неверно заполняет форму для изменения данных о товаре. Программное обеспечение выводит сообщение о неверном заполнение данных.

2.3.4 изменение статуса заказа – менеджер заказов управляет заказами, меняет их статус.

Основной поток:

- менеджер заказов переходит в панель администратора;
- система формирует панель администратора;
- менеджер заказов переходит в раздел «заказы»;
- система отображает страницу заказов;
- менеджер заказов просматривает заказы, переходит на страницу «подробнее»;
- система формирует страницу с информацией о заказе;
- менеджер изменяет статус заказа и нажимает «сохранить»;
- система регистрирует изменение статуса заказа, обновляя информацию о нём в базе данных, при статусе «доставлен».

Альтернативный поток:

- при указании неверного статуса заказа, например указание статуса выше более чем на одну стадию, или ниже данного, система выводит сообщение о неверном заполнении данных.

2.3.5 вход в аккаунт – зарегистрированный пользователь нажимает на кнопку входа в аккаунт, после чего вводит логин и пароль своей учётной записи.

Основной поток:

- пользователь нажимает на кнопку входа;
- система формирует страницу входа в аккаунт;
- пользователь указывает свои данные (логин и пароль), далее нажимает кнопку входа в аккаунт;
- система проверяет введённые данные с существующими записями в базе данных и совершает вход пользователя в программное обеспечение.

Альтернативный поток:

- пользователь неверно указывает данные для входа в аккаунт. Программное обеспечение выводит сообщение о неверном заполнении данных, после чего пользователь повторно вводит исправленные данные для входа в аккаунт.

2.3.6 регистрация аккаунта – гость создаёт свою учётную запись на странице «регистрация».

Основной поток:

- гость нажимает на кнопку регистрации;
- система формирует страницу регистрации;
- гость указывает свои данные (логин и пароль, адрес электронной почты), далее нажимает кнопку входа в аккаунт;
- система проверяет введённые данные с существующими записями в базе данных, если пользователей с совпадающими данными не существует,

то система создаёт нового пользователя в базе данных и формирует страницу входа в аккаунт.

Альтернативный поток:

- пользователь неверно указывает данные для входа в аккаунт.

Программное обеспечение выводит сообщение о неверном заполнение данных, после чего пользователь повторно вводит исправленные данные для входа в аккаунт.

2.3.7 фильтрация и поиск товаров – пользователь может найти нужный ему товар с помощью фильтрации каталога. Это можно сделать с помощью строки поиска, или с помощью фильтрации каталога.

Основной поток:

- пользователь на странице каталога нажимает на строку поиска, после чего вводит название или нажимает на категорию товаров в каталоге, находящемся в левой части экрана;

- система проводит фильтрацию каталога, выводя на главную панель перечень товаров из базы данных, проходящие фильтрацию.

Альтернативный поток:

- если нет товаров, подходящих под фильтрацию, система выводит на главную панель сообщение об отсутствии подходящих товаров в каталоге.

2.3.8 добавление сотрудников – администратор базы данных имеет возможность добавлять новых сотрудников.

Основной поток:

- администратор базы данных, переходит в панель администратора, нажимает на кнопку добавления или просмотра списка сотрудников;

- система формирует панель администратора;

- администратор базы данных, нажимает на кнопку добавления или просмотра списка сотрудников;

- система формирует страницу добавления или списка сотрудников;

– для добавления администратор вводит данные нового сотрудника и нажимает кнопку добавить;

– система регистрирует изменение пользователей и обновляет данные пользователей в базе данных.

Альтернативный поток:

– администратор неверно указывает данные для регистрации сотрудника. Программное обеспечение выводит сообщение о неверном заполнении данных, после чего администратор повторно вводит исправленные данные для регистрации.

2.3.9 обращение пользователя в поддержку – пользователь может обратиться за помощью в поддержку пользователей.

Основной поток:

– авторизованный пользователь нажимает на поле в навигации «поддержка» и переходит на страницу обращения в поддержку;

– система формирует страницу и форму обращения в поддержку;

– пользователь указывает сообщение, на которое поддержка должна ответить, и отправляет данные на сервер, нажимая на кнопку «отправить запрос».

– система регистрирует отправку нового обращения и добавляет данные в базу данных.

Альтернативный поток:

– пользователь не заполнил поле текста запроса в данном случае система отображает сообщение о неверном заполнении формы.

2.3.10 Ответ на обращение пользователя в поддержку – менеджер должен отправлять ответы на запросы пользователей на странице «запросы пользователей».

Основной поток:

– менеджер нажимает на поле в навигации «запросы пользователей» и переходит на страницу обращения в поддержку;

- система формирует страницу обращений пользователей в поддержку;
- менеджер нажимает на выбранный им запрос;
- система формирует страницу и форму ответа на запрос пользователя;
- менеджер заполняет поле ответа на запрос, и отправляет данные на сервер, нажимая на кнопку «ответить на запрос».
- система регистрирует отправку нового ответа на обращения и добавляет данные в базу данных, убирая данное обращение со страницы «запросы пользователей».

Альтернативный поток:

- менеджер не заполнил поле ответа в данном случае система отображает сообщение о неверном заполнении формы.

2.4 Описание данных

В магазине здорового питания необходимо хранить информацию о продукции, продажах, пользователях, администраторах базы данных и о менеджерах заказов. Выделяются следующие сущности:

- товар;
- заказ;
- пользователь;
- запрос;
- сотрудник;

Описание сущности «Product» представлено в таблице 2.2.1.

Таблица 2.2.1 – Описание данных сущности «Product»

Поле	Тип	Описание	Примечание
1	2	3	4
Product_Id	Числовой	Id продукта	Первичный ключ, состоит из 4 цифр
Product_Name	Текстовой	Наименование продукта	Максимальный размер 50 символов
Product_Type	Текстовой	Тип продукта	Максимальный размер 25 символов
Product_Cost	Числовой	Стоимость товара	Не меньше 0, с учётом копеек
Product_Status	Текстовой	Присутствие товара на складе	Имеет значение «В наличии» или «Не в наличии»
Product_Description	Текстовой	Описание товара	Максимальный размер 500 символов
Product_Image	Текстовой	Ссылка на файл картинки товара	Максимальный размер 25 символов

Описание сущности «Order» представлено в таблице 2.2.2.

Таблица 2.2.2 – Описание данных сущности «Order»

Поле	Тип	Описание	Примечание
1	2	3	4
Order_Id	Числовой	Id заказа	Первичный ключ, состоит из не менее чем из 1 цифры
Order_Date	Дата	Дата и время оформления заказа	По умолчанию ставится текущая дата
Order_Cost	Числовой	Стоимость заказа	Не меньше 0, с учётом копеек
Order_Address	Текстовой	Адрес доставки	Максимальное количество символов 50
Order_Phone_number	Числовой	Номер телефона, для связи с получателем	Максимальное количество символов 11

Продолжение таблицы 2.2.2

Order_Payment	Текстовой	Тип оплаты заказа	Может иметь значение «Через сайт банка», «Картой при получении», «Наличными при получении»
Order_Status	Текстовой	Статус заказа	Может иметь значение «Не оформлен», «Принят на обработку», «Сбор заказа», «Доставляется» и «Выполнен»

Описание сущности «Employee» представлено в таблице 2.2.3.

Таблица 2.2.3 – Описание данных сущности «Employee»

Поле	Тип	Описание	Примечание
1	2	3	4
Employee_Id	Числовой	Id сотрудника	Первичный ключ, состоит из 3 цифр
Employee_Job_Title	Текстовой	Должность сотрудника	Может иметь значение: «Администратор БД», или «Менеджер заказов»

Продолжение таблицы 2.2.3

Employee_Name	Текстовой	ФИО сотрудника	Максимальный размер 50 символов
Employee_Login	Текстовой	Логин сотрудника	Максимальный размер 25 символов
Employee_Password	Текстовой	Пароль сотрудника	Максимальный размер 25 символов

Описание сущности «User» представлено в таблице 2.2.4.

Таблица 2.2.4 – Описание данных сущности «User»

Поле	Тип	Описание	Примечание
1	2	3	4
User_Id	Числовой	Id пользователя	Первичный ключ, состоит из 3 цифр
User_Name	Текстовой	Имя пользователя	Максимальное количество символов – 50
User_Email	Текстовой	Электронная почта пользователя	Максимальное количество символов – 50
User_Login	Текстовой	Логин пользователя	Максимальное количество символов – 30
User_Password	Текстовой	Пароль пользователя	Максимальное количество символов – 25

Описание сущности «Request» представлено в таблице 2.2.4.

Таблица 2.2.4 – Описание данных сущности «Request»

Поле	Тип	Описание	Примечание
1	2	3	4
Request_Id	Числовой	Id запроса	Первичный ключ, состоит из 3 цифр
User_Id	Текстовой	Id пользователя	Максимальное количество символов – 50
User_Email	Текстовой	Электронная почта пользователя	Максимальное количество символов – 50
Request_Text	Текстовой	Текста запроса пользователя	Максимальное количество символов – 1000
Answer_Text	Текстовой	Текста ответа администратора	Максимальное количество символов – 1000

2.5 Логическая структура базы данных

Логическое проектирование базы данных – это преобразование модели базы данных, основанной на конкретной модели данных, в логическую модель, независимую от какой-либо СУБД. Логическое проектирование описывает организацию данных с точки зрения их логической структуры и связей между сущностями и данными в базе данных.

Логическая структура базы данных описывает схему данных, которая определяет какие данные будут храниться в базе данных, как они организованы и как между собой будут установлены связи.

Логическая структура базы данных интернет-магазина «НатурПродукт» представлена на рисунке 2.3.1.

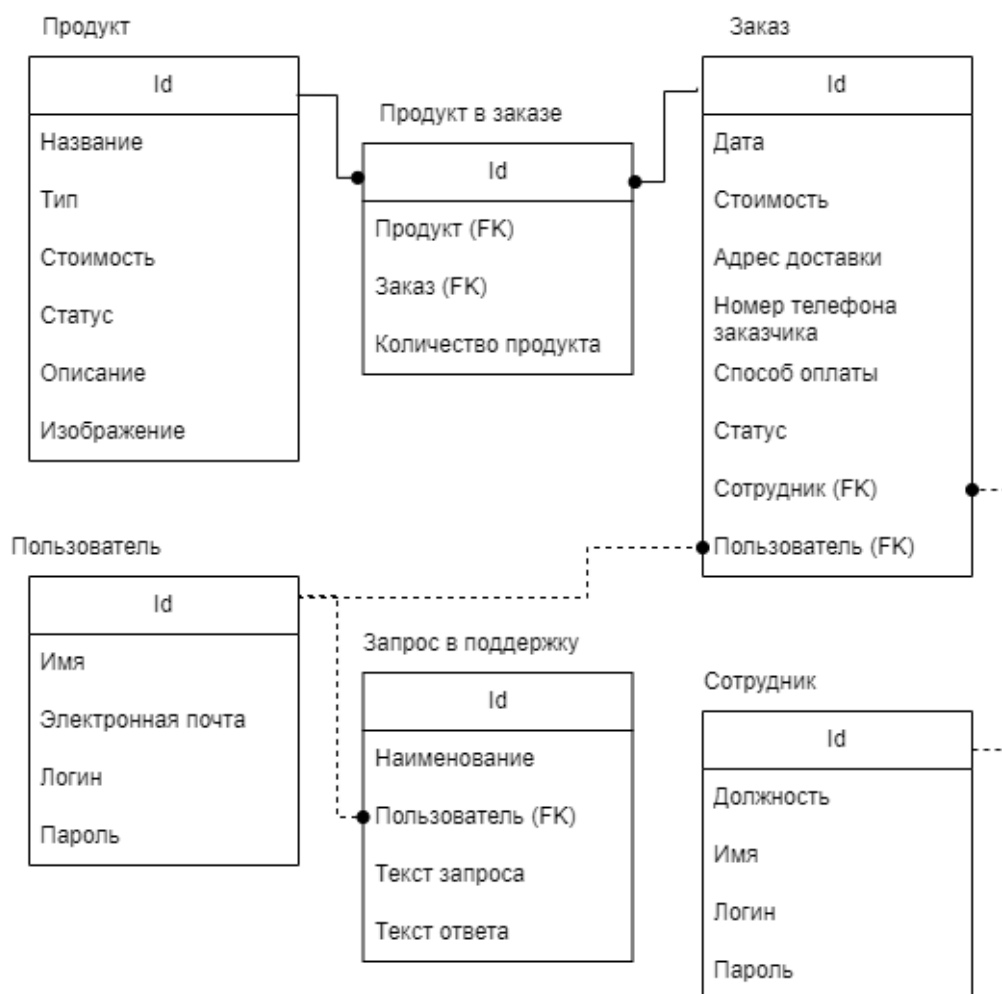


Рисунок 2.3.1 – Логическая структура базы данных «НатурПродукт»

Сущность Employee нужна для хранения информации о сотрудниках магазина.

Сущность Order нужна для хранения информации о заказах.

Сущность Request нужна для хранения информации о запросах о консультации.

Сущность Product нужна для хранения информации о товарах в магазине.

Сущность User нужна для хранения информации о пользователях.

Сущность Order_Product является ассоциативной в следствии образования связи многие ко многим между сущностями Order и Product, так

как один продукт может участвовать во многих заказах, так и в одном заказе может быть множество продуктов.

Связь между сущностями Employee и Order является одним ко многим, так как один сотрудник может участвовать во множестве заказов.

Связь между сущностями User и Order является одним ко многим, так как один пользователь может участвовать во множестве заказов.

Связь между сущностями User и Request является одним ко многим, так как один пользователь может иметь много запросов о консультации.

В данной логической схеме базы данных все связи являются не идентифицирующими, кроме связей между ассоциативной Order_Product, так как все остальные сущности являются независимыми от других сущностей.

3 Технические средства

3.1 Выбор состава технических и программных средств

Для разработки программного обеспечения и базы данных для туристического агентства были выбраны: среда разработки Pycharm, в которой был использован язык программирования Python, фреймворк Django, система управления реляционными базами данных SQLite, также HTML и CSS для размещения на веб-странице различных элементов.

Django – это фреймворк для веб-разработки, который предоставляет инструменты и структуру для создания веб-приложений на языке программирования Python. Django был разработан с целью упростить и ускорить процесс создания веб-приложений, предоставляя готовые решения для многих распространенных задач, связанных с веб-разработкой.

Преимущества Django:

административная панель: Django поставляется с встроенной административной панелью, которую можно легко настроить для управления данными и администрирования приложения без необходимости создавать пользовательский интерфейс;

шаблоны и статика: Django позволяет организовать структуру шаблонов и обработку статических файлов, что упрощает разработку пользовательского интерфейса.;

безопасность: Django включает в себя множество встроенных механизмов безопасности, таких как защита от атак инъекции SQL и других типов атак.

PyCharm – это интегрированная среда разработки (IDE) для языка программирования Python, разработанная компанией JetBrains. Она предоставляет разработчикам множество инструментов и функций, которые упрощают и улучшают процесс создания и поддержки приложений на Python. PyCharm имеет следующие преимущества:

интеграция с Git: PyCharm совмещен с системой контроля версий Git, что позволяет управлять исходным кодом и записывать изменения непосредственно из IDE;

мощный редактор кода: PyCharm предоставляет богатый редактор кода с автодополнением, подсветкой синтаксиса, проверкой ошибок и другими инструментами, которые упрощают написание и редактирование Python-кода.

Python – это высокоуровневый интерпретируемый язык программирования, который был создан в конце 1980-х годов и развивается до сих пор. Python отличается чистым и читаемым синтаксисом, что делает его одним из наиболее популярных языков программирования в мире. Он имеет следующие преимущества:

простота и читаемость: Синтаксис Python разработан так, чтобы быть легко читаемым и понятным. Это делает язык доступным для начинающих разработчиков и упрощает поддержку и сопровождение кода.

кроссплатформенность: Python поддерживается на различных операционных системах, включая Windows, macOS и различные дистрибутивы Linux, что делает его переносимым между платформами;

динамическая типизация: Python предоставляет динамическую типизацию, что упрощает разработку и уменьшает количество кода, необходимого для решения задач.

SQLite – это компактная и встроенная система управления реляционными базами данных, которая позволяет хранить, управлять и извлекать данные в локальных или встроенных в приложение базах данных. SQLite имеет следующие преимущества: простота в использовании и компактность.

HTML – это язык разметки, используемый для создания структуры и содержания веб-страниц. С помощью HTML разработчики определяют, какие элементы должны присутствовать на веб-странице и как они должны взаимодействовать друг с другом.

CSS – это язык стилей, который используется для определения внешнего вида и макета веб-страниц. CSS позволяет разработчикам задавать цвета, шрифты, размеры, отступы, расположение элементов и другие стилевые атрибуты, которые делают веб-страницу более привлекательной и удобной для пользователей.

3.2 Физическая структура базы данных

Физическая модель базы данных — это модель данных, которая определяет, каким образом хранятся данные в базе данных, отражает способ физического представления данных в памяти машины, и содержит все детали, необходимые СУБД для создания базы данных.

Физическая структура базы данных магазина «НатурПродукт» представлена на рисунке 3.2.1.

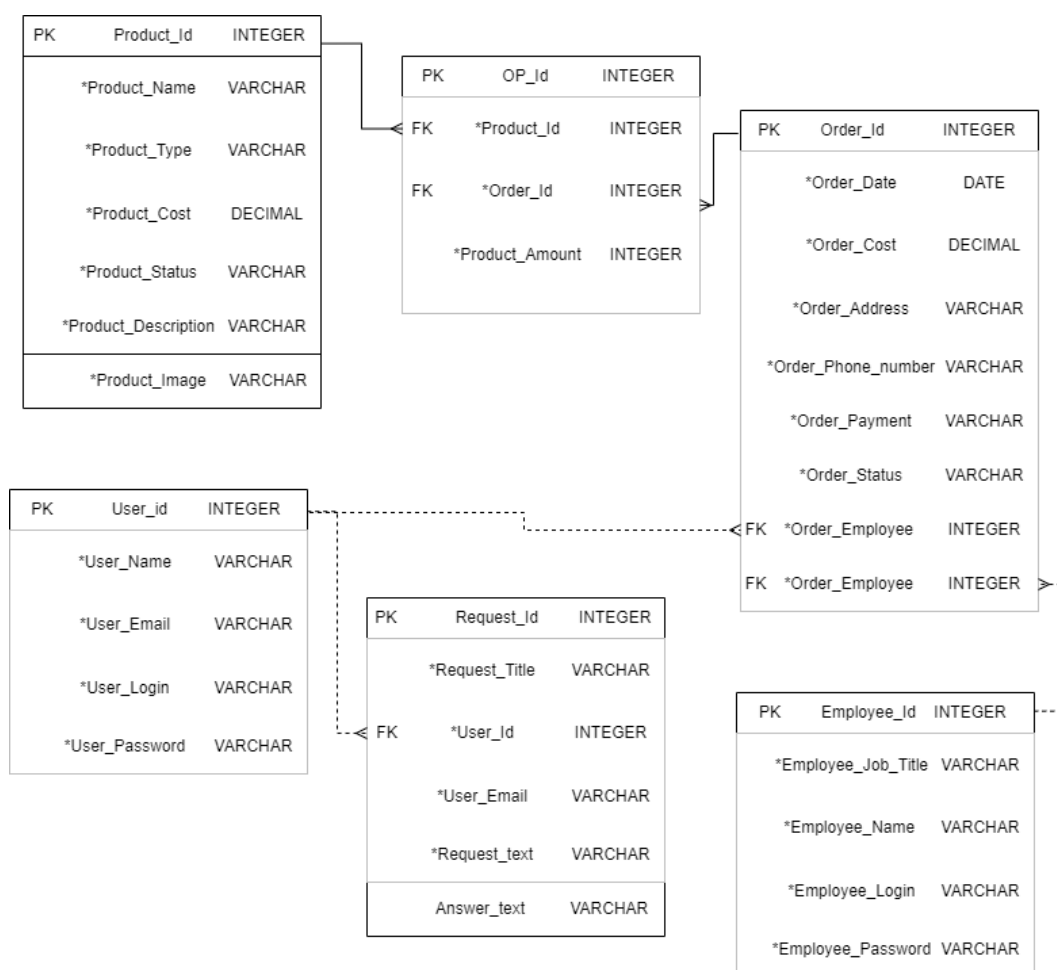


Рисунок 3.2.1 – Физическая структура базы данных «НатурПродукт»

Во всех сущностях в качестве первичного ключа выступает поле Id с типом данных INTEGER с атрибутом AUTO_INCREMENT. Атрибут AUTO_INCREMENT позволяет автоматически генерировать уникальный номер при вставке новой записи в таблицу.

Для текстовых полей выбран тип данных CHARACTER VARYING с ограничением длины записи, указанным в скобках.

Для числовых полей, таких как вес и стоимость, выбран тип данных DECIMAL. В скобках указаны ограничения по общему количеству цифр в числе и по количеству цифр после десятичной точки. Для остальных числовых полей выбран тип данных INTEGER, поскольку высокая точность не требуется.

Для полей, содержащих дату, был выбран тип данных TIMESTAMP.

За исключением поля «Answer_text» в таблице «Request», все поля являются обязательными для заполнения.

3.3 Физическая структура программы

Запуск программного обеспечения начинается с файла «manage.py», находящийся в корневой папке проекта. В этом файле также происходит автозагрузка приложений и моделей классов.

В папке «NaturProductShop» расположены файлы приложения, такие как:

- __init__.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
- asgi.py, интерфейс шлюза асинхронного сервера;
- settings.py, содержит в себе все настройки проекта. Здесь регистрируется приложения, задаётся размещение статичных файлов, настройки базы данных и так далее;
- urls.py, задаёт ассоциации url адресов с представлениями;
- wsgi.py, используется для налаживания связи между Django приложением и веб-сервером.

В папке «cart» расположены файлы приложения, такие как:

- __init__.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
- admin.py, используется для регистрации моделей в вашем приложении в администрации Django;
- apps.py, это общий файл конфигурации для всех приложений Django. С помощью этого файла можно настроить атрибуты для своего приложения;
- cart.py, содержит функции работы с сессией пользователя, для хранения в ней данных;
- forms.py, используется для определения форм Django, которые используются для ввода данных пользователем;
- models.py, в этом файле определяются модели данных Django для приложения;

- urls.py, задаёт ассоциации url адресов с представлениями;
- views.py, содержит функции и классы представления, которые обрабатывают логику запросов и ответов в приложении.

В папке «shop» расположены файлы приложения, такие как:

- __init__.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
- admin.py, используется для регистрации моделей в вашем приложении в администрации Django;
- apps.py, это общий файл конфигурации для всех приложений Django. С помощью этого файла можно настроить атрибуты для своего приложения;
- models.py, в этом файле определяются модели данных Django для приложения;
- urls.py, задаёт ассоциации url адресов с представлениями;
- views.py, содержит функции и классы представления, которые обрабатывают логику запросов и ответов в приложении.

В папке «users» расположены файлы приложения, такие как:

- __init__.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
- admin.py, используется для регистрации моделей в вашем приложении в администрации Django;
- apps.py, это общий файл конфигурации для всех приложений Django. С помощью этого файла можно настроить атрибуты для своего приложения;
- forms.py, используется для определения форм Django, которые используются для ввода данных пользователем;
- models.py, в этом файле определяются модели данных Django для приложения;
- urls.py, задаёт ассоциации url адресов с представлениями;

– views.py, содержит функции и классы представления, которые обрабатывают логику запросов и ответов в приложении.

Полная структура вышеперечисленных каталогов изображена на рисунках 3.3.1 и 3.3.2.

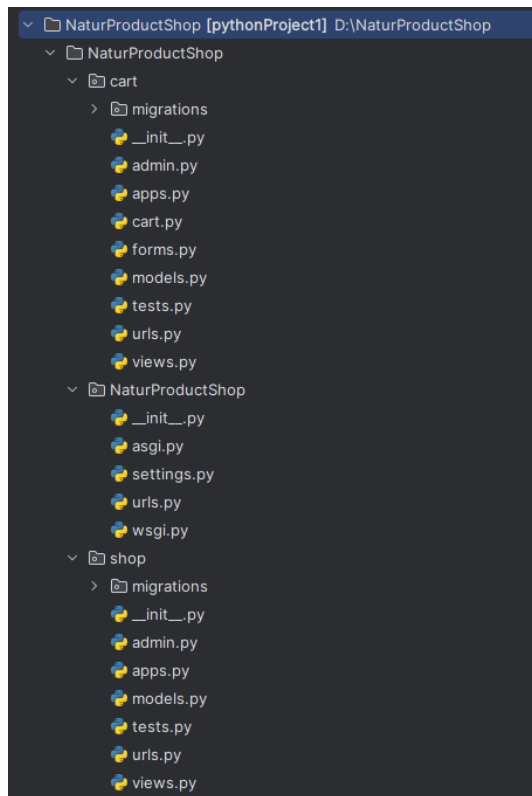


Рисунок 3.3.1 – Структура приложения

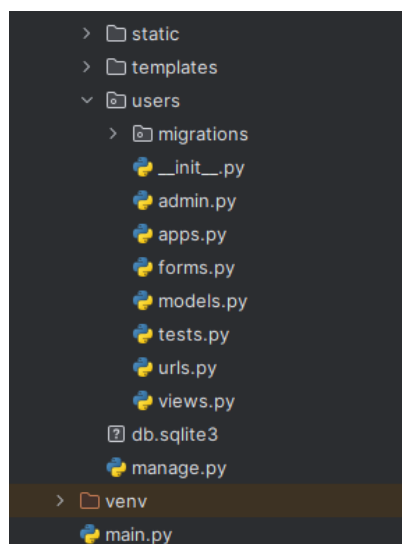


Рисунок 3.3.2 – Структура приложения

Для реализации работы с базой данных были созданы классы-модели, описывающие ее таблицы:

- Product;
- Order;
- OrderProduct;
- User;
- Employee;
- Request.

В фреймворке Django Модели классов являются автономными источниками данных, которые содержат поля классов и их методы. Каждая модель класса представляет собой таблицу из базы данных. Между моделями классов используется отношение ассоциация для установления связей.

Диаграмма классов изображена на рисунке 3.3.3.

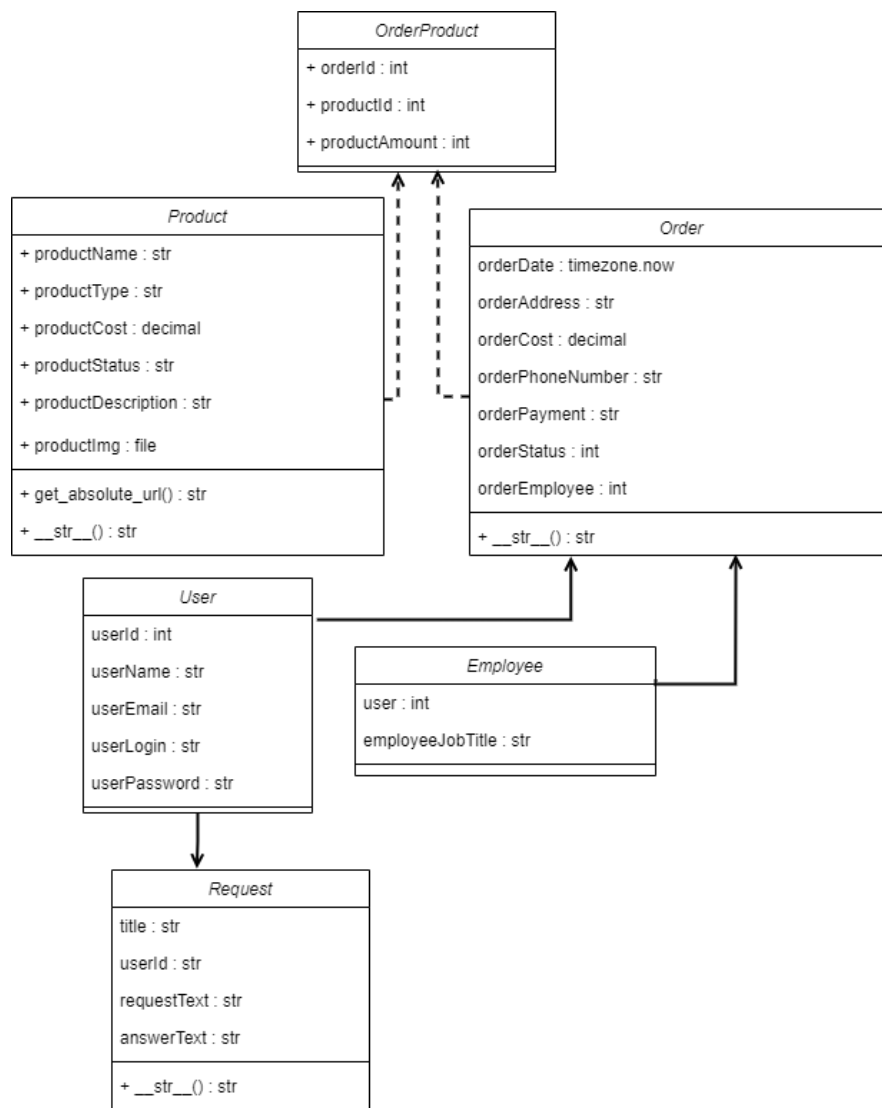


Рисунок 3.3.2 – Диаграмма классов

4 Экспериментальная часть

4.1 Тестирование приложения

Тестирование программного обеспечения было произведено методом ручного тестирования.

Метод ручного тестирования – это процесс проверки различных аспектов программного обеспечения наличием реального человека, который выполняет тестовые сценарии вручную. В процессе тестирования тестировщик взаимодействует с программным обеспечением так, как это делал бы конечный пользователь.

Тесты проводились по следующим сценариям:

4.1.1 Сценарий удачной регистрации

В форме регистрации заполняем все поля корректными данными и успешно регистрируем учетную запись.

4.1.2 Сценарий неудачной регистрации

В форме регистрации заполняем не все поля, вводим некорректные данные или вводим адрес электронной почты, который используется другим пользователем и получаем сообщения об ошибках.

4.1.3 Сценарий удачной авторизации

В форме входа ввести данные существующего пользователя, и успешно войти в учетную запись.

4.1.4 Сценарий неудачной авторизации

В форме входа ввести данные несуществующего пользователя и получить сообщение об ошибке.

					РК 09.02.03 432 06 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		32

4.1.5 Сценарий выхода из учетной записи

Нажимаем на кнопку «Выйти» в навигационном меню и успешно выходим из учетной записи.

4.1.6 Сценарий просмотра каталога продукции

На главной странице видим последние товары, на страницах категорий видим продукцию, принадлежащую выбранной категории.

4.1.7 Сценарий поиска по каталогу продукции

В строку поиска вводим поисковый запрос и находим товары, удовлетворяющие этому запросу.

4.1.8 Сценарий добавления продукции в корзину

В карточке товара нажимаем на кнопку «добавить в корзину», выводится страница корзины и видим добавленный товар. Нажимаем на кнопку увеличения или уменьшения количества товара и видим, что количество товара, его общая стоимость и общая сумма заказа изменились. Доводим количество товара до нуля и видим, что товар пропал из корзины.

4.1.9 Сценарий удачного оформления заказа

Входим в учетную запись, добавляем, как минимум один товар в корзину, переходим на страницу оформления заказа, заполняем все поля корректными данными и успешно оформляем заказ.

4.1.10 Сценарий неудачного оформления заказа

Не авторизовавшись, добавляем, как минимум один товар в корзину, переходим на страницу оформления заказа, выводится текст с информированием о нужде авторизации. Авторизовавшись и добавив, как минимум один товар в корзину, переходим на страницу оформления заказа,

заполняем не все поля или вводим некорректные данные и получаем сообщения об ошибках.

4.1.11 Сценарий удачного просмотра списка заказов

Войти в учетную запись, перейти на страницу просмотра заказов и видим все ранее оформленные заказы.

4.1.12 Сценарий неудачного просмотра списка заказов

Не авторизовавшись, вручную переходим на страницу просмотра списка заказов и выводится страница авторизации.

4.1.13 Сценарий удачного добавления пользователя

Войти в учетную запись администратора, перейти в панель администратора в раздел «Пользователи», перейти на форму добавления пользователя, заполнить все обязательные поля корректными данными, при необходимости необязательные поля также заполнить корректными данными и успешно добавить нового пользователя.

4.1.14 Сценарий неудачного добавления пользователя

Не авторизовавшись или авторизовавшись под учетной записью обычного пользователя, вручную переходим в панель администратора и выводится страница авторизации в панели администратора. Войти в учетную запись администратора, перейти на форму добавления пользователя и заполнить не все обязательные поля или вводим некорректные данные как минимум в одно поле, и получить сообщения об ошибках.

4.1.15 Сценарий удачного редактирования статуса заказа

Войти в учетную запись администратора, перейти в панель администратора в раздел «Заказы», перейти на форму редактирования заказа, выбрать необходимый статус и успешно изменить его.

4.1.16 Сценарий неудачного редактирования статуса заказа

Не авторизовавшись или авторизовавшись под обычной учетной записью, вручную переходим в панель администратора и выводится страница авторизации либо выводится главная страница соответственно.

4.1.17 Сценарий удачного добавления или редактирования продукции

Входим в учетную запись администратора, перейти в панель администратора в раздел «Продукция», перейти на форму добавления новой или редактирования существующей продукции, заполнить все обязательные поля корректными данными, при необходимости необязательные поля также заполнить корректными данными и успешно добавить или редактируем продукцию.

4.1.18 Сценарий неудачного добавления или редактирования продукции

Не авторизовавшись или авторизовавшись под учетной записью обычного пользователя, вручную перейти в панель администратора и выводится страница авторизации либо выводится страница главная страница соответственно. Войти в учетную запись администратора, перейти страницу добавления новой или редактирования существующей продукции и заполнить не все обязательные поля или ввести некорректные данные как минимум в одно поле, и получаем сообщения об ошибках.

4.1.19 Сценарий удачной отправки запроса в поддержку

Входим в учетную запись, перейти на страницу отправки заявки в поддержку, заполнить текст обращения и нажать на кнопку «отправить запрос» и успешно отправить запрос.

4.1.20 Сценарий неудачной отправки запроса в поддержку

Не авторизовавшись, перейти на страницу отправки заявки в поддержку, выводится текст с информацией о необходимости авторизации и кнопка «войти». Авторизовавшись, не заполнять текст обращения и нажать на кнопку «отправить запрос» и получить сообщение с ошибкой.

4.1.21 Сценарий удачной отправки ответа на запрос в поддержку

Войти в учетную запись менеджера, перейти на страницу просмотра запросов пользователя, затем на страницу отправки ответа на запрос, который мы выбрали, заполнить текст ответа и нажать на кнопку «отправить ответ» и успешно отправить ответ.

4.1.22 Сценарий неудачной отправки ответа на запрос в поддержку

Не авторизовавшись под учётной записью менеджера, перейти на страницу отправки ответа на запрос в поддержку, выводится текст с информацией о необходимости авторизации и кнопка «войти».

Авторизовавшись, не заполнить текст ответа на запрос и нажать на кнопку «отправить запрос» и получить сообщение с ошибкой.

4.2 Руководство пользователя

Запустив программное обеспечение, выводится главная страница. На странице содержится поле навигации, каталог товаров, поле фильтрации категории товаров, корзину и подвал сайта. Данная страница изображена на рисунках 4.2.1, 4.2.2

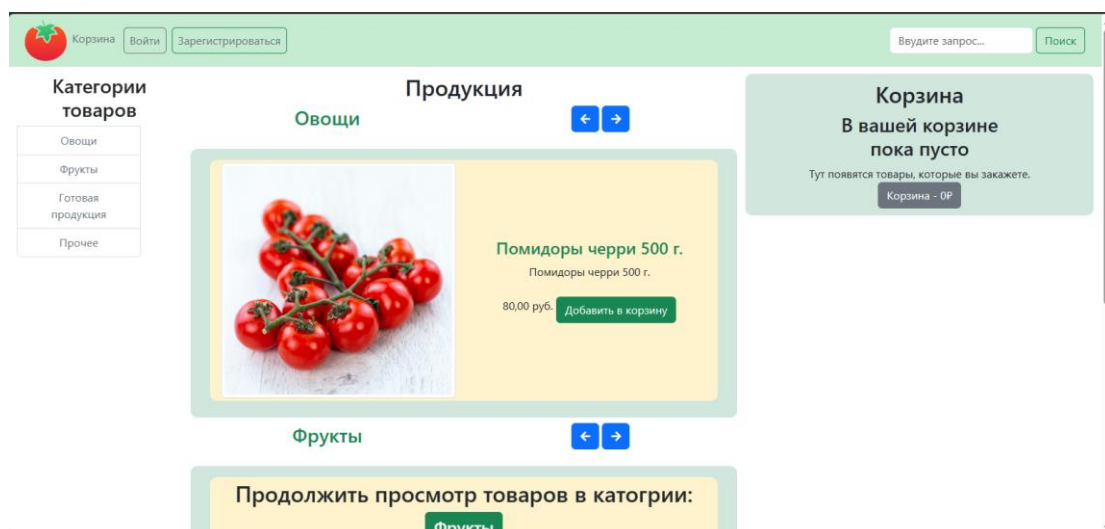


Рисунок 4.2.1 – Главная страница, раздел с навигационным меню и каруселью

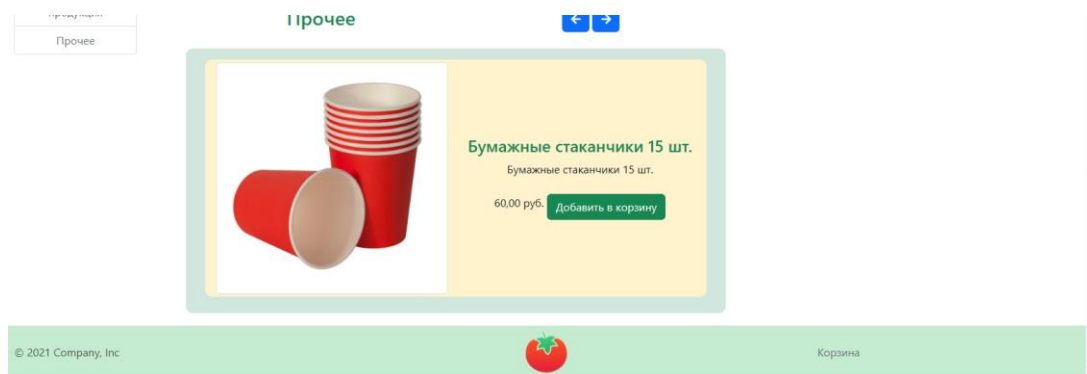


Рисунок 4.2.2 – Главная страница, раздел с подвалом сайта

Для регистрации пользователь нажимает на кнопку «Регистрация» в навигационном меню, изображенном на рисунке 4.2.1. Происходит переадресация на страницу с формой регистрации, изображенной на рисунке 4.2.3.

Изм.	Лист	№ докум.	Подпись	Дата

Рисунок 4.2.3 – Страница регистрации

Пользователь заполняет поля формы и нажимает на кнопку «Зарегистрироваться», после чего система автоматически выводит страницу регистрации пользователя, изображенную на рисунке 4.2.4.

Рисунок 4.2.4 – Страница авторизации

Пользователь заполняет поля формы и нажимает на кнопку «Войти», после чего система автоматически производит авторизацию пользователя и выводит главную страницу, изображенную на рисунке 4.2.5.

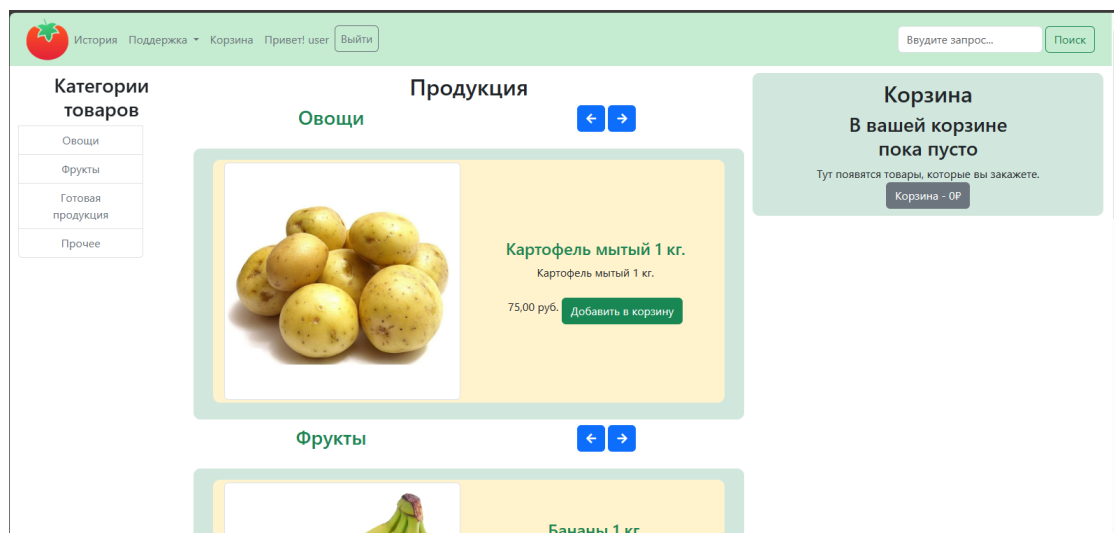


Рисунок 4.2.5 – Главная страница после авторизации

В случае, если пользователь уже имеет зарегистрированную учетную запись, он может произвести авторизацию при помощи кнопки «Вход», расположенной в навигационном меню, изображенном на рисунке 4.2.1. При нажатии на эту кнопку система переадресует пользователя на страницу с формой авторизации, изображенную на рисунке 4.2.4.

В верхней части навигационного меню расположена ссылка-изображение с логотипом предприятия предприятия. При нажатии на нее пользователь перенаправляется на главную страницу. В левой части навигационного меню расположена поисковая строка, предназначенная для поиска продукции в соответствии с поисковым запросом. Пользователь вводит текст запроса и нажимает на кнопку с надписью «Поиск», после чего система отображает продукцию, удовлетворяющую тексту поискового запроса. Пример поиска продукции изображен на рисунке 4.2.6.

Изм.	Лист	№ докум.	Подпись	Дата

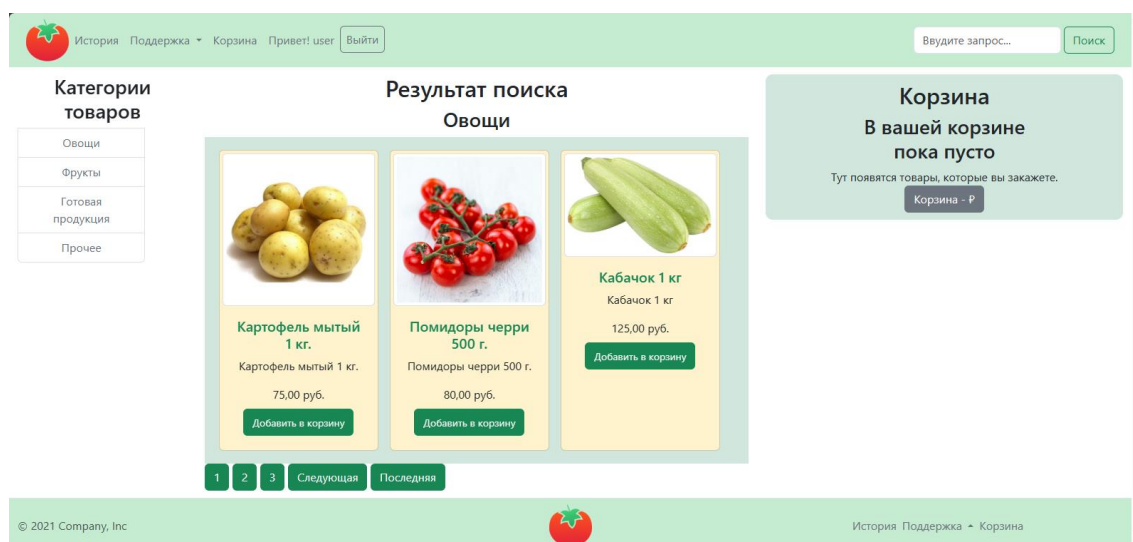


Рисунок 4.2.6 – Пример поиска продукции

В левой части навигационного меню располагаются приветственная строка и кнопка «Выйти». Для выхода из аккаунта пользователь нажимает на кнопку «Выйти», система производит разлогинивание пользователя и перенаправляет его на главную страницу, изображенную на рисунке 4.2.1. Также в левой части навигационного меню располагаются ссылки «История», «Поддержка» и «Корзина». При нажатии на ссылку «История», система переводит пользователя на страницу с историей заказов пользователя. Поскольку на данном этапе пользователь еще не совершал заказов, на странице отображается сообщение об этом с кнопкой, при нажатии на которую, система переводит пользователя на главную страницу, изображенную на рисунке 4.2.5. Страница «История» изображена на рисунке 4.2.7.

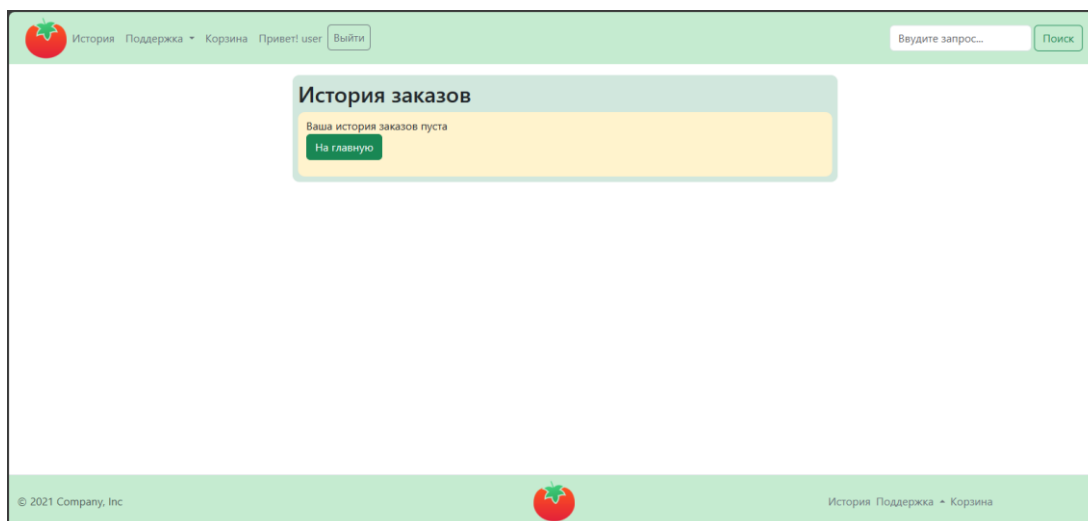


Рисунок 4.2.7 – Пустая страница «История»

При нажатии на ссылку «Поддержка» появляется выпадающее меню, содержащее в себе пункты «Отправить запрос» и «Ваши запросы» (рисунок 4.2.8).

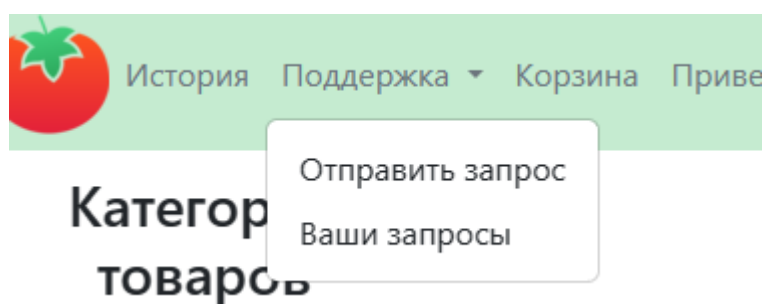


Рисунок 4.2.8 – Выпадающее меню ссылки «Поддержка»

Для перехода на страницу отправки запроса в поддержку, пользователь нажимает на пункт «Отправить запрос», после чего система перенаправляет пользователя на страницу с формой отправки запроса (рисунок 4.2.9).

Изм.	Лист	№ докум.	Подпись	Дата

Рисунок 4.2.9 – Страница отправки запроса в поддержку

После заполнения поля «Текст запроса», пользователь нажимает на кнопку «Отправить запрос», после чего система сохраняет запрос на сервер и переводит пользователя на Главную страницу, изображенную на рисунке 4.2.5.

Для перехода на страницу просмотра истории запросов пользователя, пользователь нажимает на пункт «Ваши запросы», после чего система перенаправляет пользователя на страницу с историей отправленных запросов (рисунок 4.2.10).

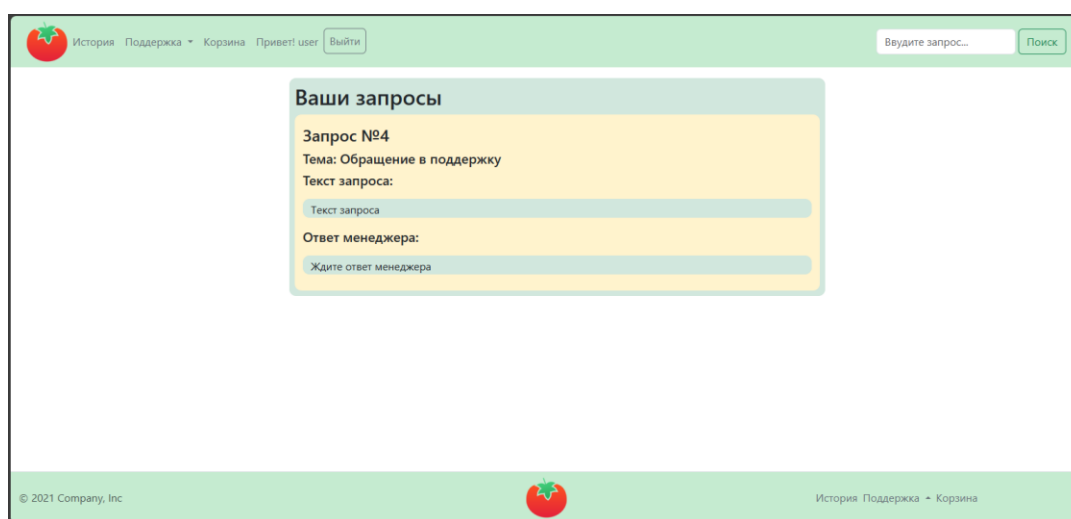


Рисунок 4.2.10 – Страница истории запросов

При нажатии на пункт меню навигации «Корзина» система перенаправляет пользователя на страницу с корзиной выбранных товаров (рисунок 4.2.11).

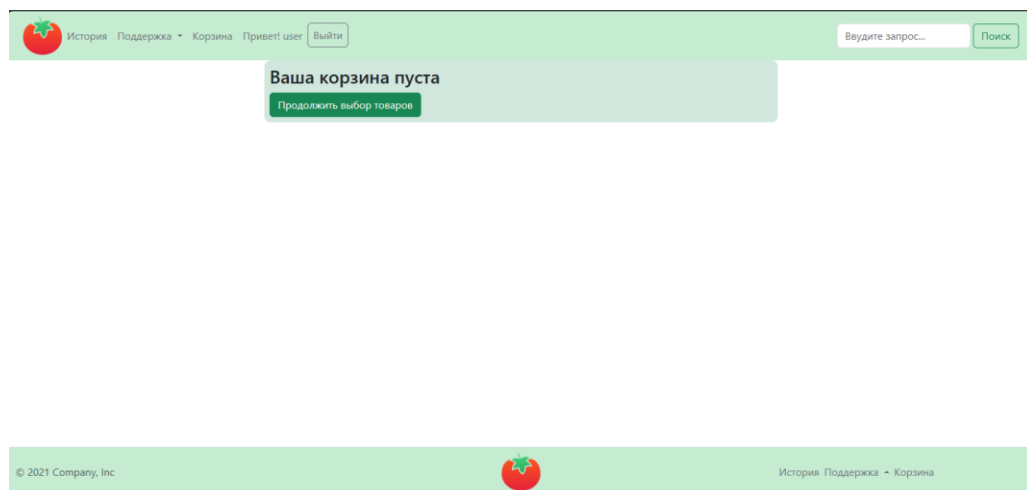


Рисунок 4.2.11 – Страница корзины

Так как пользователь ещё не добавлял товаров в корзину, на странице отображается сообщение об этом с кнопкой, при нажатии на которую, система переводит пользователя на главную страницу, изображенную на рисунке 4.2.5.

Для перехода на страницу продукции пользователь нажимает на название в карточке продукции, после чего перенаправляется на страницу выбранной продукции (рисунок 4.2.12).

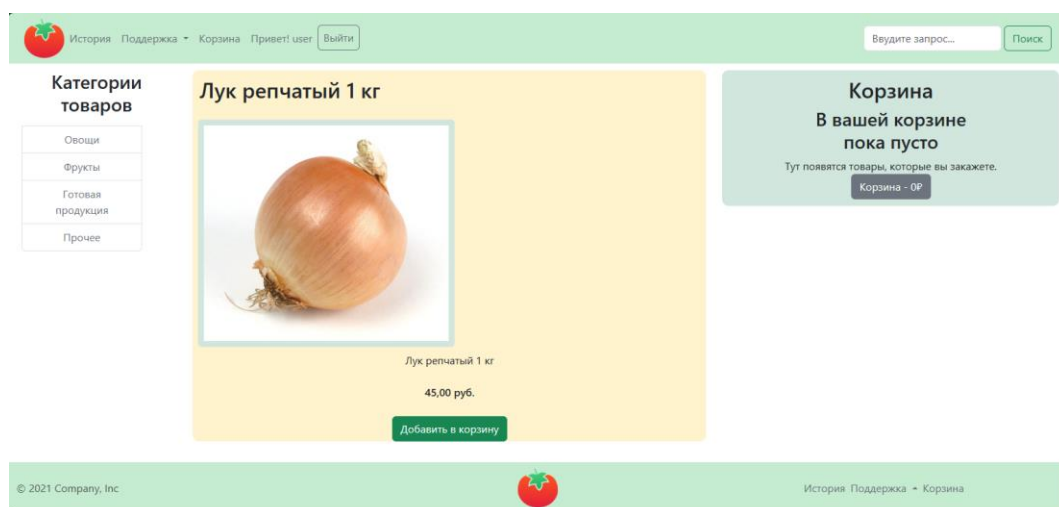


Рисунок 4.2.12 – Страница продукции

Для добавления продукции в корзину пользователь нажимает на кнопку «Добавить в корзину» в карточке продукции или на странице продукции, после чего выбранная продукция добавляется в корзину пользователя, а пользователь перенаправляется на страницу корзины (рисунок 4.2.13).

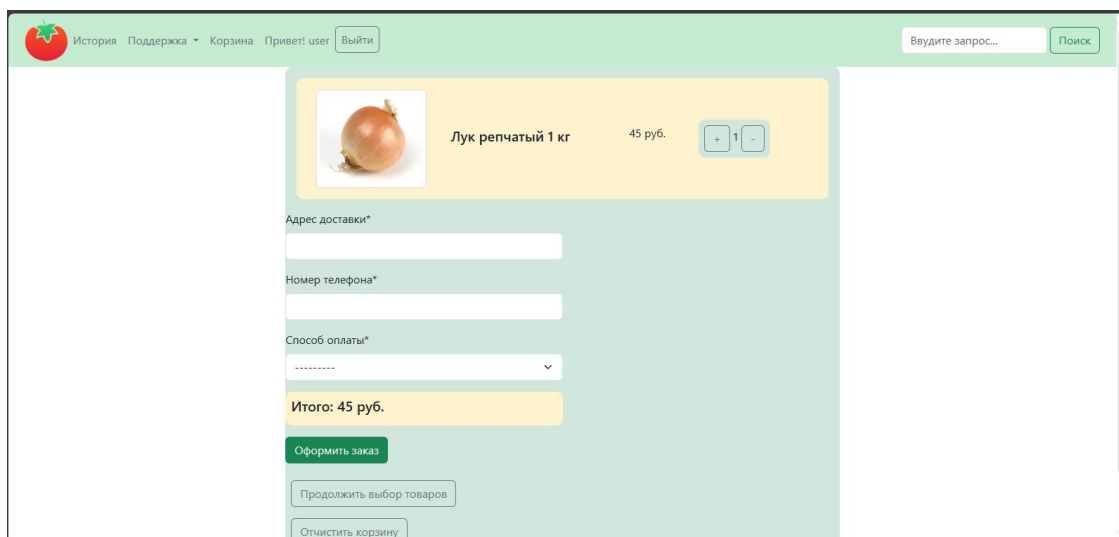


Рисунок 4.2.13 – Товар добавлен в корзину

Кнопками «+» и «-» пользователь может увеличивать или уменьшать количество выбранного товара в корзине. Для оформления заказа пользователь заполняет все поля в форме оформления заказа, далее нажимает на кнопку «Оформить заказ», после чего система сохраняет заказ в базе данных и перенаправляет пользователя на главную страницу, изображенную на рисунке 4.2.5.

Если пользователь авторизовался под учетной записью администратора, тогда в навигационной панели, появляется пункт «Панель администратора» (рисунок 4.2.14).

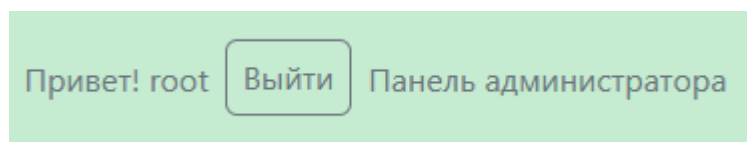


Рисунок 4.2.13 – Пункт панель администратора

Для перехода на страницу администрирования пользователь нажимает на этот пункт, после чего перенаправляется на страницу администрирования, изображенную на рисунке 4.2.15.

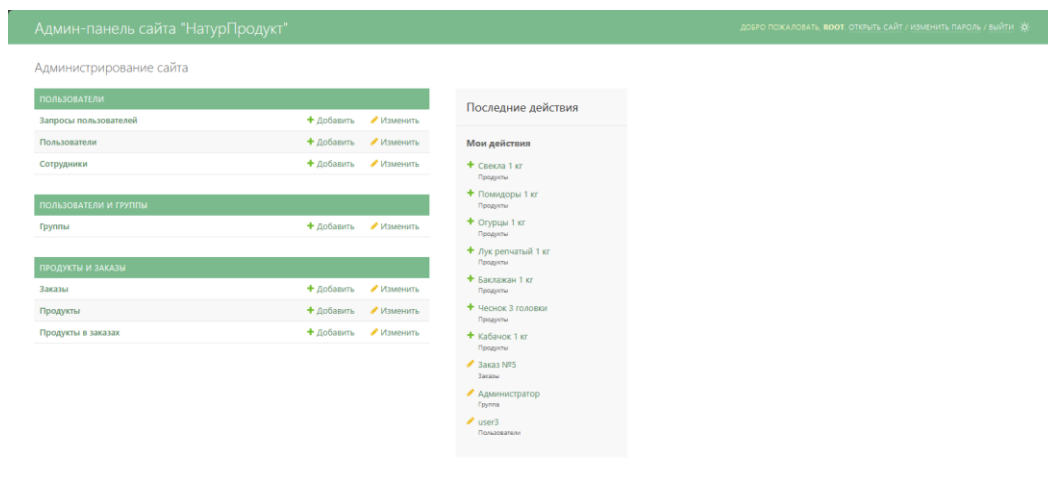


Рисунок 4.2.13 – Панель администратора

В навигационном меню слева располагаются ссылки на таблицы, доступные для взаимодействия. Для возврата на главную страницу сайта пользователь нажимает на кнопку «Открыть сайт», после чего перенаправляется на главную страницу (рисунок 4.2.5). Для просмотра данных необходимой таблицы пользователь нажимает на соответствующую ссылку, после чего происходит перенаправление на страницу взаимодействия с данными выбранной таблицы. В качестве примера на рисунке 4.2.14 изображена страница взаимодействия с заказами.

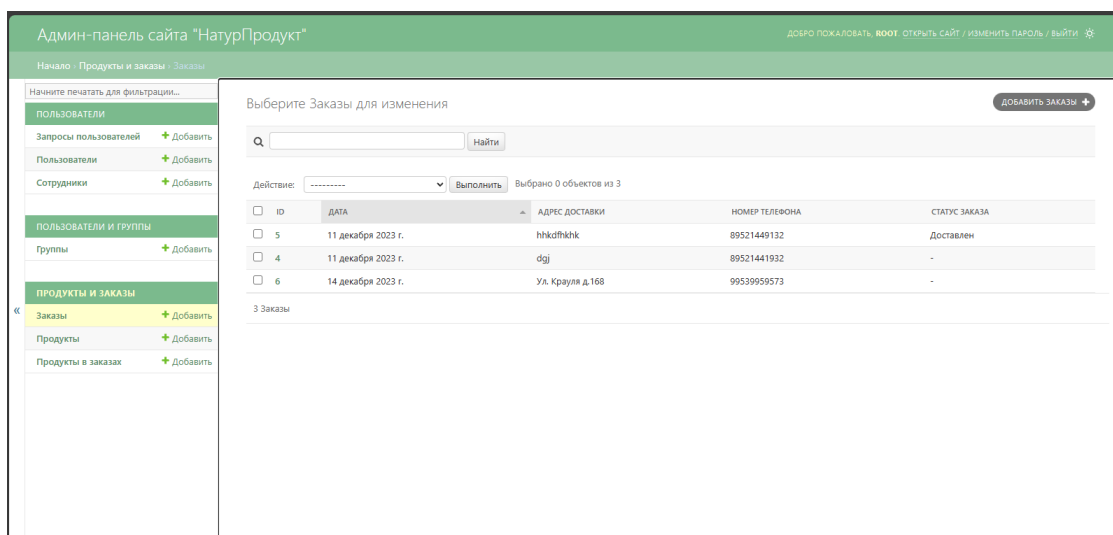


Рисунок 4.2.14 – Страница взаимодействия с заказами

Для добавления новой записи пользователь нажимает на кнопку «Добавить заказы», после чего перенаправляется на форму добавления. Данная форма есть у таблиц «Заказы», «Пользователи», «Продукты»,

«Сотрудники». Пользователь вводит данные в форму и нажимает на кнопку «Сохранить», после чего перенаправляется на предыдущую страницу. Форма добавления на примере таблицы «Заказы» изображена на рисунке 4.2.15.

Рисунок 4.2.15 – Страница создания

Для редактирования записи пользователь нажимает на номер заказа, после чего перенаправляется на форму редактирования. Данная форма есть у таблиц «Заказы», «Продукция», «Сотрудники». Пользователь вводит данные в форму и нажимает на кнопку «Сохранить», после чего перенаправляется на предыдущую страницу. Форма редактирования на примере таблицы «Заказы» изображена на рисунке 4.2.16.

Рисунок 4.2.16 – Страница редактирования заказа

Заключение

В ходе выполнения курсовой работы, цель которой заключалась в проектировании и разработке программного обеспечения для магазина здорового питания «НатурПродукт», был проведен анализ предметной области и информации, используемой в базе данных, были выявлены основные потребности бизнеса и пользователей, что послужило основой для разработки вариантов использования программного обеспечения и логической структуры базы данных.

Спроектированная физическая структура базы данных и программного обеспечения позволила создать эффективное и масштабируемое решение, удовлетворяющее потребностям магазина здорового питания «НатурПродукт».

Разработанное программное обеспечение обладает интуитивно понятным и удобным для пользователя интерфейсом. Внедрение программного обеспечения для предприятия «НатурПродукт» обеспечивает современные инструменты для управления его бизнес-процессами, повышает эффективность работы и способствует развитию предприятия.

Проведенное тестирование подтвердило надежность и стабильность программного обеспечения, а разработанное руководство пользователя позволит пользователям быстро освоить функционал программы.

Таким образом, результаты курсового проекта представляют собой ценный вклад в развитие магазина здорового питания «НатурПродукт». Внедрение разработанного программного продукта на предприятие может значительно повысить конкурентоспособность магазина на рынке. Из всего вышеперечисленного следует, что заявленные задачи были выполнены, поставленная цель была достигнута.

Список использованных источников

1. Сайт Django Project [Электронный ресурс]. – Режим доступа: <https://docs.djangoproject.com/en/5.0/topics/forms/formsets/>;
2. Сайт Bootstrap [Электронный ресурс]. – Режим доступа: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>;
3. Сайт StudFiles [Электронный ресурс]. – Режим доступа: <https://studfile.net/preview/5910972/page:6/>;
4. Сайт METANIT [Электронный ресурс]. – Режим доступа: <https://metanit.com/python/django/5.9.php>;
5. Сайт GitBook [Электронный ресурс]. – Режим доступа: <https://ilyachch.gitbook.io/django-rest-framework-russian-documentation/overview/readme>.