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. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_ А. В. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  | Разработчик  \_\_\_\_\_\_\_\_\_\_\_\_\_А. А. Голосов  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  |  |

2023

**Министерство общего и профессионального образования Свердловской области**

**Государственное автономное профессиональное образовательное**

**учреждение Свердловской области**

**«Уральский радиотехнический колледж им. А.С. Попова»**

**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
   1. Описание предметной области
   2. Диаграммы вариантов использования
   3. Сценарии вариантов использования
   4. Описание данных
   5. Логическая структура базы данных 3 Технический проект
   6. Выбор состава технических и программных средств
   7. Физическая структура базы данных
   8. Физическая структура программы
3. Экспериментальная часть
   1. Тестирование приложения
   2. Руководство пользователя.

Заключение

Список использованных источников

Дата выдачи « » 20 г

Срок окончания « » 20 г

Преподаватель А.В. Беляева

Содержание

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

РК 09.02.03 423 06 ПЗ

Разраб.

Голосов А. А.

Провер.

Беляева А. В.

Н. Контр.

Беляева А. В.

Проектирование и разработка программного обеспечения для магазина здорового питатния "НатурПродукт"

Лит.

Листов

ГАПОУ CO

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. В нынешнее время существуют технологи 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.

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.

# 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 понимая 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 (неавторизированный пользователь);
* покупатель;
* менеджер заказов;
* администратор базы данных;

Функции гостя:

* просмотр каталога товаров;
* поиск товаров по фильтрации;
* 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.
* вход в аккаунт;
* выход из аккаунта.

Функции менеджера заказов:

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

# Системный проект

# 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 дождаться 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 документы о проведении данных операций, в которых находятся данные о заказе (продукты в заказе, их количество, стоимость, номер заказа) и о клиенте (имя, номер телефона, адрес доставки). Кроме этого, существуют документы, которые напрямую влияют на статистику магазина, например, чек о продаже (продукты в заказе, стоимость заказа, дата, номер заказа).

1. 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:

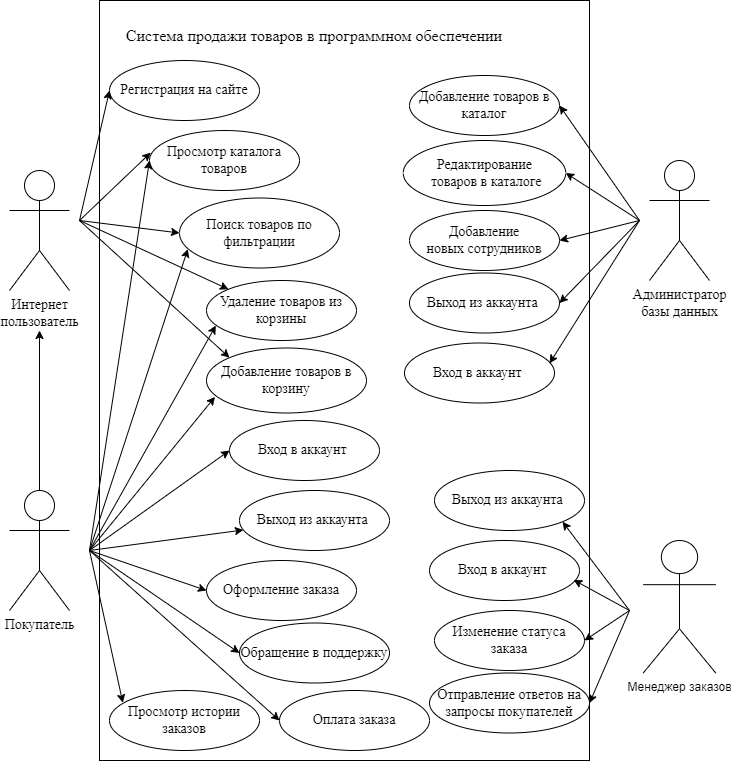


Рисунок 2.2.1 – Диаграмма вариантов использования

1. Сценарии вариантов использования

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

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 авторизирован.

1. 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».

1. 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.

1. 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 вывод сообщение о неверном заполнении данных.

1. 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 заполнение данных, после чего пользователь повторно вводит исправленные данные для входа в аккаунт.

1. 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.

1. 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 выводи на главную панель сообщение об отсутствии подходящих товаров в каталоге.

1. 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 заполнение данных, после чего администратор повторно вводит исправленные данные для регистрации.

1. Обращение пользователя в поддержку – пользователь может обратиться за помощью в поддержку пользователей.

Основной поток:

* авторизированный 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.

1. 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.

# 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 |

# Логическая структура базы данных

Логическое проектирование базы данных – это преобразование модели базы данных, основанной на конкретной модели данных, в логическую модель, независящую 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.

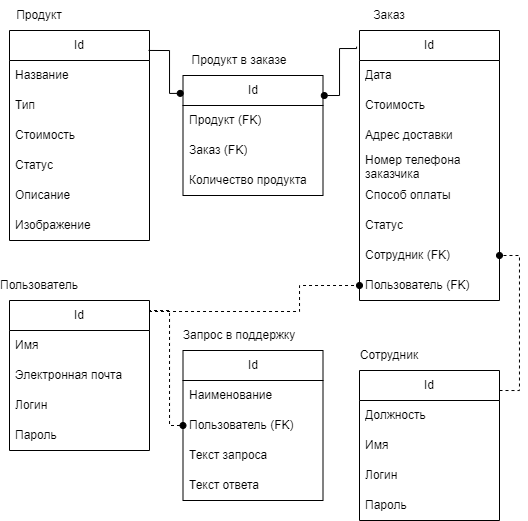


Рисунок 2.3.1 – Логическая структура базы данных «НатурПродукт»

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 одном ко многим, так как один сотрудник может участвовать во множестве заказов.

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

Связь между сущностями User и Request является одном 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.

# Технические средства

# 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 интегрирован с системой контроля версий Git, что позволяет управлять исходным кодом и коммитить 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 компакность.

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.

# 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.

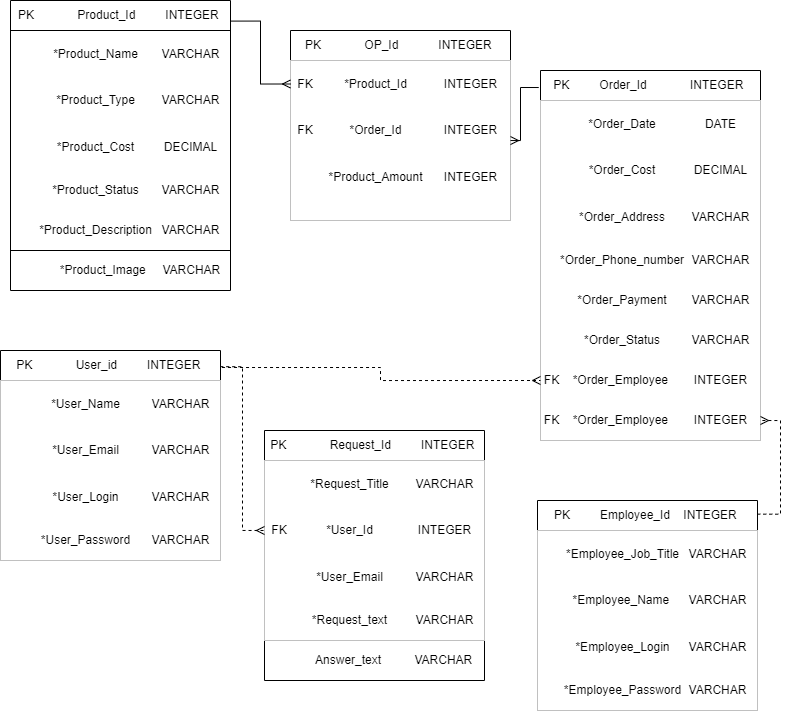


Рисунок 3.2.1 – Физическая структура базы данных «НатурПродукт»

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.

1. 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 файле «NaturProductShop/shop» расположены файлы приложения, такие как:

* \_\_init\_\_;
* admin;
* apps;
* models;
* urls;
* views.

В файле «NaturProductShop/cart» расположены файлы приложения, такие как:

* \_\_init\_\_;
* admin;
* cart;
* apps;
* forms;
* models;
* urls;
* views.

В файле «NaturProductShop/NaturProductShop» расположены файлы приложения, такие как:

* \_\_init\_\_;
* asgi;
* urls;
* wsgi.

В файле «NaturProductShop/users» расположены файлы приложения, такие как:

* \_\_init\_\_;
* admin;
* apps;
* forms;
* models;
* urls;
* views.

Полная структура вышеперечисленных каталогов изображена на рисунках 3.3.1 и 3.3.2.

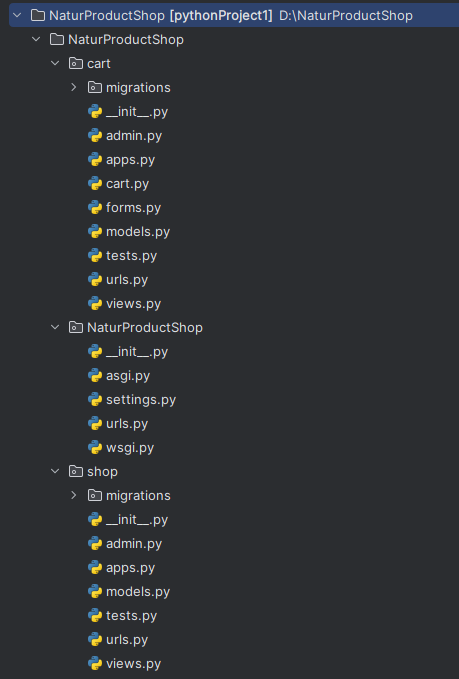


Рисунок 3.3.1 – Структура приложения

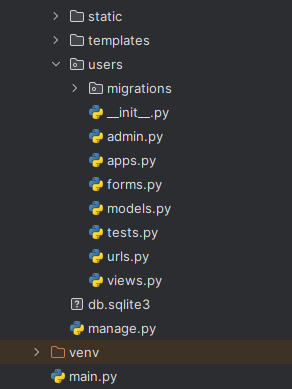


Рисунок 3.3.2 – Структура приложения

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.

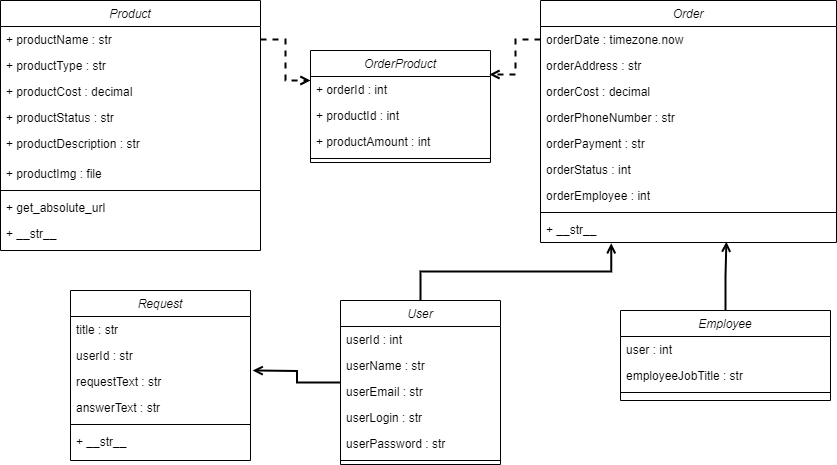


Рисунок 3.3.2 – Диаграмма классов

1. Экспериментальная часть
2. 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:

1. Сценарий удачной регистрации

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

1. 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.

1. Сценарий удачной авторизации

В форме входа вводим данные существующего пользователя, и успешно входим в учетную запись.

1. Сценарий неудачной авторизации

В форме входа вводим данные несуществующего пользователя и получаем сообщение об ошибке.

1. Сценарий выхода из учетной записи

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

1. Сценарий просмотра каталога продукции

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

1. Сценарий поиска по каталогу продукции

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

1. 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.

1. 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.

1. 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.

1. Сценарий удачного просмотра списка заказов

Входим в учетную запись, переходим на страницу просмотра своих заказов и видим все ранее оформленные заказы.

1. Сценарий неудачного просмотра списка заказов

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

1. Сценарий удачного изменения данных учетной записи

Входим в учетную запись, переходим на страницу настроек учетной записи, в необходимое поле вводим корректные данные и успешно изменяем информацию учетной записи.

1. Сценарий неудачного изменения данных учетной записи

Не авторизовавшись, пытаемся вручную перейти на страницу настроек учетной записи и переадресуемся на страницу авторизации. Входим в учетную запись, переходим на страницу настроек учетной записи и не заполняем необходимое поле или вводим некорректные данные, и получаем сообщение об ошибке.

1. Сценарий удачного добавления пользователя

Входим в учетную запись администратора, переходим в панель администратора в раздел «Пользователи», переходим на форму добавления пользователя, заполняем все обязательные поля корректными данными, при необходимости необязательные поля также заполняем корректными данными и успешно добавляем нового пользователя.

1. Сценарий неудачного добавления пользователя

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

1. Сценарий удачного редактирования статуса заказа

Входим в учетную запись администратора, переходим в панель администратора в раздел «Заказы», переходим на форму редактирования заказа, выбираем необходимый статус и успешно изменяем его.

1. Сценарий неудачного редактирования статуса заказа

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

1. Сценарий удачного добавления или редактирования продукции

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

1. Сценарий неудачного добавления или редактирования продукции

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

1. Сценарий удачной отправки запроса в поддержку

Входим в учетную запись, переходим на страницу отправки заявки в поддержку, заполняем текст обращения и нажимаем на кнопку «отправить запрос» и успешно отправляем запрос.

1. Сценарий неудачной отправки запроса в поддержку

Не авторизовавшись вручную переходим на страницу отправки заявки в поддержку, выводится текст с информацией о необходимость авторизации и кнопка «войти». Авторизовавшись, не заполняем текст обращения и нажимаем на кнопку «отправить запрос» и видим сообщение с ошибкой.

1. Сценарий удачной отправки ответа на запрос в поддержку

Входим в учетную запись менеджера, переходим на страницу отправки ответа на запрос в поддержку, заполняем текст ответа и нажимаем на кнопку «отправить ответ» и успешно отправляем ответ.

1. Сценарий неудачной отправки ответа на запрос в поддержку

Не авторизовавшись под учётной записью менеджера, вручную переходим на страницу отправки ответа на запрос в поддержку, выводится текст с информацией о необходимость авторизации и кнопка «войти». Авторизовавшись, не заполняем текст ответа на запрос и нажимаем на кнопку «отправить запрос» и видим сообщение с ошибкой.

# Заключение

В данной курсовой работе была разработана база данных и приложение для автоматизации процессов внутри предприятия «НатурПродукт». Целью работы было создание удобного инструмента для управления деятельностью магазина, его продажами и сбором статистики. В ходе работы были поставлены задачи, такие как описание предметной области, изучение данных, определение логической структуры базы данных и создание технического проекта.

Разработанная база данных сопровождается программным приложением, которое обеспечивает удобное взаимодействие с базой данных. Преимущества данного программного продукта заключаются в его простоте использования, отсутствии необходимости особых навыков у пользователя и возможности загрузки базы данных на другое устройство для повышения мобильности.

Внедрение данной системы позволяет ускорить работу и автоматизировать множество процессов, которые ранее выполнялись вручную. Это приводит к повышению точности и актуальности данных, а также упрощению работы с учётом продукции, ведением продаж и формированием статистики магазина.

Таким образом, с разработкой базы данных и программного обеспечения для предприятия «НатурПродукт» достигается значительный прогресс в автоматизации и оптимизации его бизнес-процессов. Внедрение данной системы позволит предприятию эффективно управлять своей деятельностью, повысить производительность и обеспечить безопасность информации.

В результате разработки базы данных и программного обеспечения для предприятия «НатурПродукт» были достигнуты следующие положительные результаты:

* эффективное управление данными. Внедрение базы данных позволило предприятию «НатурПродукт» эффективно организовать хранение и управление своими данными. База данных предоставляет структурированное хранение информации о товарах, клиентах, продажах, возвратах и других бизнес-процессах, что облегчает доступ к необходимым данным и их обработку;
* повышение производительности. Автоматизация множества процессов, таких как учёт товаров, оформление продаж, возвраты и составление отчётности, позволяет сократить время выполнения операций и повысить общую производительность предприятия. Сотрудники тратят меньше времени на рутинные задачи, освобождая больше времени для выполнения стратегически важных задач;
* обеспечение безопасности информации. База данных защищена от введения некорректных данных и проведения ошибочных операций. Разделение прав доступа на администратора базы данных и пользователя минимизирует риск возникновения ошибок и вредительства данных. Это обеспечивает сохранность и конфиденциальность информации о клиентах, продукции и других бизнес-данных предприятия;
* упрощение работы и улучшение точности данных. Программное приложение, сопровождающее базу данных, имеет простой и понятный интерфейс, что упрощает работу пользователям. Автоматизация процессов и предоставление готовых шаблонов документов (например, чеков о продаже и документов о возврате) способствуют улучшению точности данных и снижению возможности ошибок, связанных с ручным вводом и обработкой информации;
* улучшение аналитики и статистики. Благодаря базе данных и сопровождающему приложению предприятие «НатурПродукт» может собирать и анализировать статистические данные о продажах, возвратах, запасах и других параметрах. Это позволяет предприятию получать ценные и точные данные для принятия управленческих решений, планирования стратегий и оптимизации бизнес-процессов.

Внедрение разработанной базы данных и программного обеспечения для предприятия «НатурПродукт» обеспечивает современные инструменты для управления его бизнес-процессами, повышает эффективность работы и способствует развитию предприятия.

# Список использованных источников

1. PostgreSQL. Основы языка SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова. — СПб.: БХВ-Петербург, 2018. — 336 с.
2. Основы технологий баз данных: учеб. пособие / Б. А. Новиков, Е. А. Горшкова, Н. Г. Графеева; под ред. Е. В. Рогова. — 2-е изд. — М.: ДМК Пресс, 2020. — 582 с.
3. Документация PostgreSQL [Электронный ресурс]. – Режим доступа: https://postgrespro.ru/docs/postgresql
4. Грошев А.С. Основы работы с базами данных [Электронный ресурс]. – Режим доступа: https://studfile.net/preview/2983990/page:13/
5. Карпова Т.С. Базы данных: модели, разработка, реализация. [Электронный ресурс]. – Режим доступа: http://www.intuit.ru/department/database/dbmdi/
6. Кузнецов С.Д. Введение в реляционные базы данных [Электронный ресурс]. – Режим доступа: https://intuit.ru/studies/courses/74/74/lecture/27919
7. Кривоносова, Н. В. Технология WPF. Разработка модулей программного обеспечения: практикум: учебное пособие / Н. В. Кривоносова. — Санкт-Петербург: СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 132 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/279719
8. Домбровская, Г. Оптимизация запросов PostgreSQL / Г. Домбровская, Б. Новиков, А. Бейликова; перевод с английского Д. А. Беликова. — Москва: ДМК Пресс, 2021. — 278 с. — ISBN 978-5-97060-963-7. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/241103
9. Прокушев, Я. Е. Базы данных: учебное пособие / Я. Е. Прокушев. — 2-е изд., доп. — Санкт-Петербург: Интермедия, 2022. — 264 с. — ISBN 978-5-4383-0250-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/217925