Министерство образования и молодежной политики Свердловской области

Государственное автономное профессиональное

образовательное учреждение Свердловской области

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

Проектирование и разработка программного обеспечения для деятельности дилерского автоцентра «АвтоМоська»

Пояснительная записка к курсовому проекту

по ПМ.03 Участие в интеграции программных модулей

по МДК 03.01 Технология разработки программного обеспечения

РК 09.02.03.432 03 ПЗ

|  |  |  |
| --- | --- | --- |
| Нормоконтролер  \_\_\_\_\_\_\_\_\_\_\_ А.В. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  | Разработчик  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Г.О. Биринцев  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  |  |

Екатеринбург

2023

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

[1 Постановка задачи 5](#_Toc147317680)

[2 Системный проект 7](#_Toc147317681)

[2.1 Описание предметной области 7](#_Toc147317682)

[2.2 Диаграммы вариантов использования 8](#_Toc147317683)

[2.3 Сценарии вариантов использования 10](#_Toc147317684)

[2.4 Описание данных 13](#_Toc147317685)

[2.5 Логическая структура базы данных 19](#_Toc147317686)

[3 Технический проект 22](#_Toc147317687)

[3.1 Оформление выбора технических и программных средств 22](#_Toc147317688)

[3.2 Физическая структура 23](#_Toc147317689)

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

РК 09.02.03 432 03 ПЗ

Разраб.

Биринцев Г. О.

Провер.

Беляева А. В.

Н. Контр.

Беляева А. В.

Проектирование и разработка программного обеспечения для деятельности дилерского автоцентра «АвтоМоська»

Лит.

Листов

22

ГАПОУ CO

УРТК им. А.С. Попова

Введение

Основанием для проектирования и разработки программного обеспечения для дилерской компании ООО «АвтоМоська» является необходимость упрощения работы с покупателями, удобного оформления заявок, которые составляет покупатель самостоятельно через программное обеспечение.

Проект является актуальным в связи с современными потребностями дилерской компании в автоматических информационных системах, без которых не будет возможна полноценная конкуренция с более технически развитыми компаниями в сфере продаж автомобилей и оказания смежных услуг.

Проект предоставит возможность:

* улучшения бизнес-процессов – сбор данных, которые анализируются для улучшения деятельности компании и увеличения прибыли;
* управления данными о клиентах – отслеживание активности клиентов и хранение контактных данных;
* сокращения времени и обеспечения легкого доступа к программе (общее хранилище данных дилерского центра, реализованное в одной платформе);
* улучшения работы между отделами дилерской компании, передача данных между отделами продаж и обслуживания, хранение информации о прошлых покупках.

Система управления базами данных имеет решающее значение для эффективной работы и развития бизнеса компании.

В приложении будет реализован весь необходимый функционал. Это предоставление разных уровней доступа для разделения полномочий в управлении приложением. Менеджер будет иметь доступ к чтению и редактированию основной информации о товарах, услугах, клиентах.

# Постановка задачи

Целью курсового проекта является разработка программного обеспечения для автоматизации процессов внутри предприятия дилерского центра «АвтоМоська», для удобного управления деятельностью центра, которая позволит эффективно хранить, обрабатывать и анализировать информацию о продаже автомобилей, услугах сервисного обслуживания, запасных частях и финансовых операциях, а также управлять доступом к этой информации.

Для достижения поставленной цели необходимо выполнить следующие задачи:

* изучить предметную область и процессы внутри предприятия, определить, какие процессы можно автоматизировать с помощью базы данных;
* разработать диаграмму вариантов использования. Изучить и проанализировать данные отражающие роли и задачи предметной области;
* разработать сценарии вариантов использования. Изучить и описать вероятные сценарии использования программного обеспечения;
* описать данные, которые будут храниться в базе данных, на основе изучения процессов и потребностей предприятия;
* разработать логическую структуру базы данных, определить способы обмена информацией с приложением и методы хранения данных;
* создать технический проект, включающий в себя разработку базы данных с учетом выявленных потребностей, определение необходимых технических и программных средств, которые позволят эффективно управлять деятельностью компании, её продажами и сбором статистики.

Разделы пояснительной записки к курсовому проекту:

* в разделе «постановка задачи» будет определена цель курсового проекта и поставлены задачи для достижения данной цели;
* в разделе «системный проект» будет описана структура работы базы данных и приложения, их логическая структура;
* в разделе «технический проект» будет выбран состав технических и программных средств для разработки базы данных и приложения и описана физическая структура базы данных;
* в разделе «экспериментальная часть» будет проведено тестирование созданного программного обеспечения и базы данных, после которого будет описано руководство пользователя, которое обеспечит пользователя объяснением всего функционала программного обеспечения у каждого из пользователя системы.

# Системный проект

## Описание предметной области

Дилерский автоцентр «Автомоська» является официальным дилером многих известных автомобильных марок. Он предоставляет услуги продажи новых автомобилей, а также оказывает услуги по сервисному обслуживанию и ремонту.

В дилерском автоцентре «Автомоська» можно выделить следующие бизнес-процессы:

* продажа автомобилей. Этот процесс включает в себя следующие шаги: прием заявки на автомобиль, консультация покупателя по характеристикам и моделям автомобилей (консультант-эксперт по автомобилям), демонстрация автомобиля, проведение тест-драйва (тест-драйв-менеджер), оформление документов на покупку (договор купли-продажи, счет-фактура, акт приема-передачи), продажа дополнительных услуг и аксессуаров, передача автомобиля покупателю (акт приема-передачи);
* обслуживание автомобилей. Этот процесс включает в себя прием заявок на обслуживание (заявка на ремонт или ТО), запись на сервис (журнал учета заявок), проведение диагностики (карта диагностики), ремонт и замена деталей и узлов автомобиля, проведение технического обслуживания (акт выполненных работ), уведомление клиента о готовности автомобиля, выдача автомобиля клиенту (акт приема-передачи);
* закупка запчастей и аксессуаров. Этот процесс включает в себя оценку потребности в запчастях и аксессуарах, выбор поставщиков, закупку запчастей и аксессуаров (заказ на поставку), контроль качества поставляемых товаров, хранение и учет запчастей и аксессуаров (учетный журнал);
* управление персоналом. Этот процесс включает в себя подбор (анкета кандидата), обучение (план обучения), мотивацию персонала (программа мотивации), планирование графиков работы (график работы сотрудников), контроль выполнения задач и общей производительности (отчеты о выполнении задач), решение конфликтных ситуаций;
* финансовый учет и контроль. Этот процесс включает в себя управления бизнесом. Команда бухгалтеров отвечает за сбор, анализ и отчетность данных, связанных с доходами, расходами и налоговыми платежами. Они также отслеживают финансовые операции, поддерживают взаимоотношения с налоговыми органами и готовят отчеты для управленческого комитета. В этот процесс входят документы, такие как счета, квитанции, договоры, акты и другие отчетные документы;
* маркетинг и продвижение. Этот процесс заключается в популяризации продукта, привлечения новых клиентов и увеличения продаж. Маркетинговый отдел разрабатывает стратегию продвижения продукта на рынок, определяет целевую аудиторию и проводит исследования рынка. Команда маркетологов занимается созданием рекламных материалов, в том числе брошюр, листовок, рекламных роликов, и разрабатывает кампании в социальных сетях и других медиа-платформах.

## Диаграммы вариантов использования

Диаграммы вариантов использования — это визуальные модели, которые помогают описать, как различные пользователи взаимодействуют с системой, выполняя определенные действия. Для дилерского автоцентра «АвтоМоська» представлена следующая диаграмма вариантов использования, представленная на рисунке 2.2.1:

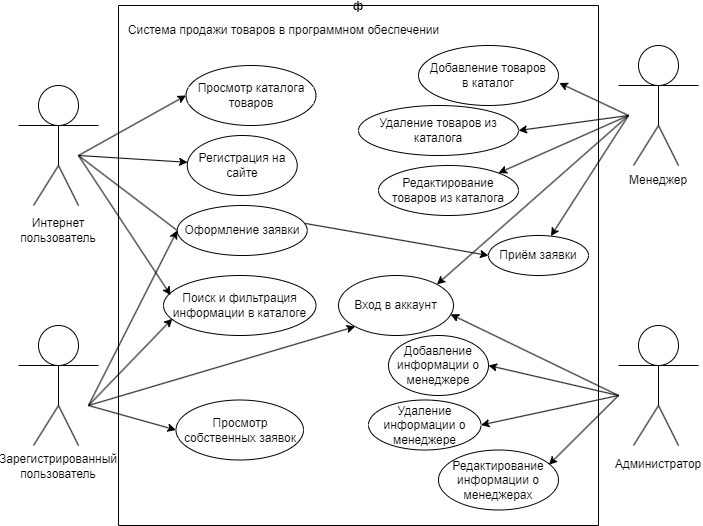


Рисунок 2.2.1 – Диаграмма вариантов использования

Возможные действия интернет пользователя, который не авторизирован:

* просмотр каталога товаров;
* оформление заявки на товар;
* регистрация на сайте.

Доступные действия покупателя:

* вход в аккаунт через ввод логина и пароля;
* оформление заявки на товар;
* просмотр существующих заявок.

Доступные действия администратора базы данных:

* добавление информации о новых менеджерах;
* удаление информации о менеджерах;
* редактирование информации о менеджерах;

Доступные действия менеджера заказов:

* связь с покупателем;
* изменение статуса заказа;
* добавление товаров в каталог;
* удаление товаров из каталога;
* редактирование товаров в каталоге.

## Сценарии вариантов использования

Для дилерского центра существуют различные сценарии использования:

1. оформление заявки – покупатель хочет оформить заявку на приобретение автомобиля.

Основной поток событий:

* cистема загружает каталог автомобилей;
* покупатель просматривает и выбирает автомобиль в каталоге, после нажимает на кнопку «Оформить заявку», где он заполняет поля: ФИО, город, регион, почта, телефон, выбранный автомобиль;
* система создаёт заявку в базе данных и выводит информацию о её подтверждении на главную страницу.

Альтернативные потоки:

* необходимый товар недоступен. Менеджер указывает статус товара в базе данных «недоступен в данный момент»;
* покупатель неверно заполнил форму заявки. Программное обеспечение проверяет правильность заполнения формы заявки, при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

* для оформления заявки пользователь обязательно должен быть авторизированным.

1. Просмотр собственных заявок – покупатель хочет посмотреть историю своих заявок.

Основной поток событий:

* покупатель открывает историю своих заявок нажав на кнопку «История заявок», где указан текущий статус заявок и полная информация о них;
* система берёт данные по покупателю и его заявкам из базы данных и выводит их списком;
* покупатель может выбрать из списка конкретный автомобиль нажав на кнопу «Просмотр», чтобы увидеть информацию об этой заявке;
* система берёт данные о заявке из базы данных и выводит их покупателю.

Альтернативные потоки:

* пользователь не верно указывает данные для регистрации или входа в аккаунт. Программное обеспечение выводит сообщение о неверном заполнение данных, а также предлагает восстановить пароль, если у пользователя уже существует своя учётная запись.

Предусловия:

* для просмотра истории пользователь обязательно должен быть авторизированным;
* для восстановления пароля пользователь должен иметь свой аккаунт.

1. Добавление товаров в каталог – менеджер хочет добавить автомобиль в каталог.

Основной поток событий:

* менеджер нажимает на кнопку «Добавить», где он заполняет поля: Наименование, Бренд.
* система внесёт введённую информацию в базу данных.

Альтернативный поток:

* менеджер не верно указывает формат данных для заполнения полей. Программное обеспечение выводит сообщение о неверном заполнение данных, а также предложит повторный ввод и укажет на ошибки.

Предусловия:

* для добавления информации об автомобиле менеджер обязательно должен быть авторизирован;

1. Удаление товаров из каталога – менеджер хочет убрать автомобиль из каталога.

Основной поток событий:

* менеджер нажимает на кнопку «Удалить».
* система удалит информацию о выбранном автомобиле из базы данных.

Предусловия:

* для удаления информации об автомобиле менеджер обязательно должен быть авторизирован;

1. Редактирование товаров из каталога – менеджер хочет редактировать информацию об автомобиле.

Основной поток событий:

* менеджер нажимает на кнопку «Редактировать», где он меняет содержимое полей.
* система обновит введённую информацию в базе данных.

Альтернативный поток:

* менеджер не верно указывает формат данных для заполнения полей. Программное обеспечение выводит сообщение о неверном заполнение данных, а также предложит повторный ввод и укажет на ошибки.

Предусловия:

* для редактирования информации об автомобиле менеджер обязательно должен быть авторизирован;

1. Приём заявки – менеджер хочет принять заявку от покупателя.

Основной поток событий:

* менеджер нажимает на кнопку «Изменить статус заявки», где он меняет статус заявки клиента: в ожидании, в обработке, в процессе передачи, выполнен.
* система обновит введённую информацию в базе данных.

Предусловия:

* для изменения информации о статусе заявки менеджер обязательно должен быть авторизирован;

1. Добавление информации о менеджере – администратор хочет добавить нового менеджера.

Основной поток событий:

* администратор нажимает на кнопку «Добавить», где он заполняет поля: ФИО,;
* система внесёт введённую информацию в базу данных.

Альтернативный поток:

* администратор не верно указывает формат данных для заполнения полей. Программное обеспечение выводит сообщение о неверном заполнение данных, а также предложит повторный ввод и укажет на ошибки.

Предусловия:

* для добавления информации о менеджере, администратор обязательно должен быть авторизирован;

1. Удаление информации о менеджере – администратор хочет убрать информацию о менеджере.

Основной поток событий:

* менеджер нажимает на кнопку «Удалить».
* система удалит информацию о выбранном автомобиле из базы данных.

Предусловия:

* для удаления информации об автомобиле менеджер обязательно должен быть авторизирован;

1. Редактирование товаров из каталога – менеджер хочет редактировать информацию об автомобиле.

Основной поток событий:

* менеджер нажимает на кнопку «Редактировать», где он меняет содержимое полей.
* система обновит введённую информацию в базе данных.

Альтернативный поток:

* менеджер не верно указывает формат данных для заполнения полей. Программное обеспечение выводит сообщение о неверном заполнение данных, а также предложит повторный ввод и укажет на ошибки.

Предусловия:

* для редактирования информации об автомобиле менеджер обязательно должен быть авторизирован;

1. управление заказами – менеджер заказов управляет заявками, меняет их статус, отменяет, или делает возврат.

Основной поток:

* менеджер заказов просматривает и обрабатывает заявку.
* программное обеспечение генерирует чек и отчет о заказе.

Альтернативный поток:

покупатель решает отменить заказ или сделать возврат. Менеджер заказов рассматривает заявку на отмену заказа или оформление возврата. Для проведения возврата продукции покупателю необходимо предоставить электронную выписку из банка о оплате заказа, за которую покупатель хочет вернуть деньги, а менеджеру заказов необходимо принять данный запрос о возврате, сделать вывод о полноценности этого запроса и совершить возврат, если доказательства о необходимости возврата буду исчерпывающими, или попросить дополнительные сведения, или вовсе отклонить запрос о возврате.

1. обработка оплаты – при оформлении заказа покупатель выбирает способ оплаты (безналично через сайт банка, безналично при доставке, наличными при доставке)

Основной поток:

* покупатель выбирает метод оплаты при оформлении заказа. Возможна оплата безналично через сайт банка, картой при доставке, или наличными при доставке.
* программное обеспечение обрабатывает платеж, если оплата происходит через сайт банка.
* если оплата картой или наличными при получении, то оплата происходит с помощью курьера, который передаёт информацию об успешной оплате менеджеру заказов.

Альтернативный поток:

* оплата через сайт банка невозможна. В этом случае программное обеспечение выведет сообщение с предложением об оплате картой или наличными при получении.
* оплата при получении невозможна. В данном случае курьер передаёт информацию о том, что покупатель не может оплатить заказ и менеджер заказов отменяет данный заказ.

1. консультация покупателя и менеджера заказов – покупатель или пользователь с помочью онлайн-чата могут начать диалог.

Основной поток:

* покупатель или администратор заказов начинает онлайн-чат, с целью обсуждения вопросов.

1. регистрация на сайте и вход в аккаунт –
2. фильтрация и поиск товаров –
3. добавление новых администраторов и менеджеров -

Эти сценарии вариантов использования помогут лучше понять, как различные пользователи и системы взаимодействуют с дилерским центром «АвтоМоська» и какие основные функциональности он предоставляет.

## Описание данных

Для базы данных дилерского автоцентра необходимо хранить информацию о клиентах, сотрудниках, услугах, товарах и финансовой составляющей. Выделяются следующие объекты:

* клиент;
* сотрудник (администратор, менеджер);
* услуга;
* товар;
* автомобили;
* финансовые операции.

Описание сущности «Клиент» представлено в таблице 2.1.

Таблица 2.1 – Описание данных сущности «Client»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id клиента | Первичный ключ |
| FirstName | Текстовый | Имя клиента | Заполнение обязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| LastName | Текстовый | Фамилия клиента | Заполнение необязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| Patronymic | Текстовый | Отчество клиента | Заполнение необязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| Phone | Текстовый | Телефон клиента | Заполнение обязательно, только цифры, длина 11, уникальное значение |

Описание сущности «Сотрудник» представлено в таблице 2.2.

Таблица 2.2 – Описание данных сущности «Worker»

| Поле | Тип | Описание | Примечание |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id сотрудника | Первичный ключ |
| FirtsName | Текстовый | Имя сотрудника | Заполнение обязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| LastName | Текстовый | Фамилия сотрудника | Заполнение обязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| Patronymic | Текстовый | Отчество сотрудника | Заполнение необязательно, минимальная длина 2, максимальная длина 20, только буквы (кириллица) |
| Post | Текстовый | Должность сотрудника (администратор, менеджер) | Заполнение обязательно, значение из заданного списка |
| Login | Текстовый | Логин для доступа к системе | Заполнение обязательно, уникальное значение, минимальная длина 4, максимальная длина 20, только буквы (латиница) и цифры |
| Password | Текстовый | Пароль для доступа к системе | Заполнение обязательно, минимальная длина 6, максимальная длина 20,  только буквы (латиница) и цифры |
| Phone | Текстовый | Телефон сотрудника | Заполнение обязательно, только цифры, длина 11, уникальное значение |

Описание сущности «Услуга» представлено в таблице 2.3.

Таблица 2.3 – Описание данных сущности «Service»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id услуги | Первичный ключ |
| Name | Текстовый | Наименование услуги | Заполнение обязательно,  минимальная длина 2, максимальная длина 50,  только буквы (кириллица) |
| Description | Текстовый | Описание услуги | Заполнение необязательно,  минимальная длина 15, максимальная длина 200 |
| Duration | Числовой | Срок услуги | Заполнение обязательно,  Только цифры, ограничение времени от 1 часа до 128 часов |
| Price | Числовой | Стоимость услуги | Заполнение обязательно, положительное число |

Описание сущности «Товар» представлено в таблице 2.4.

Таблица 2.4 – Описание данных сущности «Product»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id товара | Первичный ключ |
| Name | Текстовый | Наименование товара | Заполнение обязательно,  минимальная длина 2, максимальная длина 50 |
| Description | Текстовый | Описание товара | Заполнение необязательно,  минимальная длина 15, максимальная длина 200 |
| Category | Текстовый | Категория товара | Заполнение обязательно,  максимальная длина 50,  минимальная длина 2,  только буквы (кириллица) |
| Price | Числовой | Стоимость товара | Заполнение обязательно, положительное число |

Описание сущности «Автомобили» представлено в таблице 2.5.

Таблица 2.5 – Описание данных сущности «Auto»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id автомобиля | Первичный ключ |
| Model | Текстовый | Модель автомобиля | Заполнение обязательно, минимальная длина 2, максимальная длина 30 |
| Brand | Текстовый | Марка автомобиля | Заполнение обязательно, минимальная длина 2, максимальная длина 30 |
| Сonstruction | Текстовый | Конструкция автомобиля | Заполнение обязательно,  максимальная длина 100 |
| Description | Текстовый | Описание автомобиля | Заполнение необязательно,  минимальная длина 15, максимальная длина 500, только буквы (кириллица) и цифры |
| Motor | Текстовый | Мотор автомобиля | Заполнение обязательно,  минимальная длина 5, максимальная длина 20 |
| Gearbox | Текстовый | Коробка передач автомобиля | Заполнение обязательно, максимальная длина 50 |
| Drive | Текстовый | Привод автомобиля | Заполнение обязательно, максимальная длина 50 |
| Color | Текстовый | Цвет автомобиля | Заполнение обязательно, минимальная длина 5, максимальная длина 20,  только буквы (кириллица) |
| Price | Числовой | Стоимость автомобиля | Заполнение обязательно, положительное число |

Описание сущности «Финансы» представлено в таблице 2.6.

Таблица 2.6 – Описание данных сущности «Finance»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| Id | Числовой | Id операции | Первичный ключ |
| Date | Дата | Дата проведения операции | По умолчанию заполняется текущая дата |
| Amount | Числовой | Сумма операции | Заполнение обязательно, положительное число |

## Логическая структура базы данных

Логическая проектирование базы данных — это описание организации данных в базе данных, которое определяет, как эти данные связаны и зависят друг от друга. Она определяет таблицы, их атрибуты и связи между ними, обеспечивая целостность и качество данных.

Это описание базы данных позволяет разработчикам и администраторам баз данных понимать, как данные организованы и как они могут быть доступны приложениям. Логическая структура обычно представляется в виде диаграммы, которая отображает сущности (таблицы), атрибуты (столбцы) и связи между ними. Логическая структура базы данных дилерского автоцентра представлена на диаграмме 2.3.1

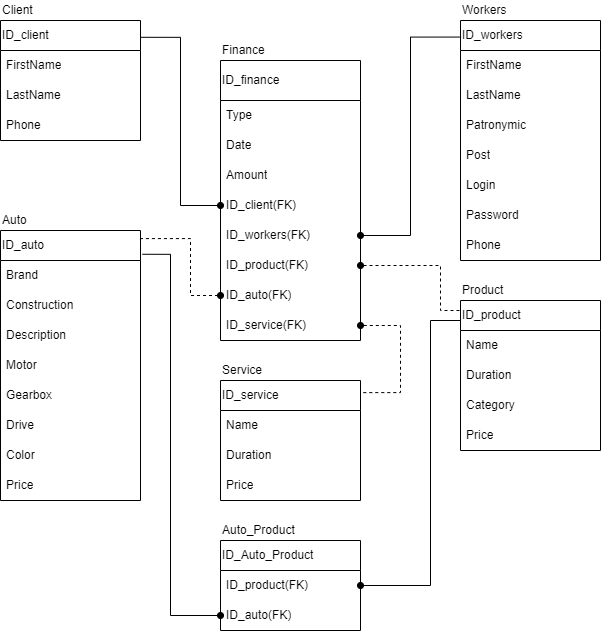


Рисунок 2.3.1 Логическая структура базы данных

Сущность Auto необходима для хранения информации об автомобилях, которые реализуются на предприятии.

Сущность Client необходима для хранения информации о клиенте дилерского автоцентра.

Сущность Workers необходима для хранения информации о сотруднике, который работает в дилерском автоцентре.

Сущность Product необходима для хранения информации об автомобильных товарах дилерского автоцентра.

Сущность Service необходима для хранения информации об услугах дилерского автоцентра.

Сущность Finance необходима для хранения информации о продажах дилерского автоцентра.

Связь между сущностями Workers и Finance выбрана один ко многим, потому что один работник может быть во многих продажах.

Связь между сущностями Product и Finance выбрана один ко многим, потому что один автомобильный товар может быть во многих продажах, либо может отсутствовать.

Связь между сущностями Сlient и Finance выбрана один ко многим, потому что клиент может быть во многих продажах.

Связь между сущностями Auto и Finance выбрана один ко многим, потому что один автомобиль может быть во многих продажах, либо может отсутствовать.

Связь между сущностями Service и Finance выбрана один ко многим, потому что одна услуга может быть во многих продажах, либо может отсутствовать.

Связь между сущностями Auto и Product выбрана многие ко многим, потому что многие автомобильные товары могут быть во многих автомобилях.

# Технический проект

## Оформление выбора технических и программных средств

Для разработки базы данных для дилерского центра были выбраны технические и программные средства, которые наилучшим образом соответствуют требованиям проекта.

Для создания базы данных была использована система управления реляционными базами данных (СУБД) PostgreSQL, которая является свободной и открытой для использования. Для управления базой данных и выполнения запросов было использовано приложение PgAdmin, которое позволяет удобно работать с PostgreSQL. Среди преимуществ можно выделить наличие графического интерфейса, возможность быстрой навигации по базе данных.

Для создания приложения была выбрана среда разработки Visual Studio, которая является одним из наиболее популярных инструментов для создания приложений под операционную систему Windows. Visual Studio обладает богатым функционалом, включая интегрированную среду разработки, отладчик, поддержку различных языков программирования и технологий. Был использован язык программирования C#, который является объектно-ориентированным языком программирования. Он обеспечивает высокую производительность и мощные возможности для работы с объектами. Он также поддерживает различные платформы и технологии, что обеспечивает гибкость в разработке приложений.

Npgsql.EntityFrameworkCore.PostgreSQL представляет собой фреймворк, способствующий взаимодействию с базой данных PostgreSQL. Основные преимущества данного фреймворка включают простоту использования, бесплатность и возможность эффективного взаимодействия с PostgreSQL.

Для работы приложения, разработанного для дилерского центра, необходимо учитывать следующие характеристики:

* операционная система: «Windows» 7 или выше;
* процессор: с тактовой частотой не менее 1 ГГц;
* оперативная память (ОЗУ): объемом не менее 2 ГБ;
* дисковое пространство: не менее 1 ГБ.

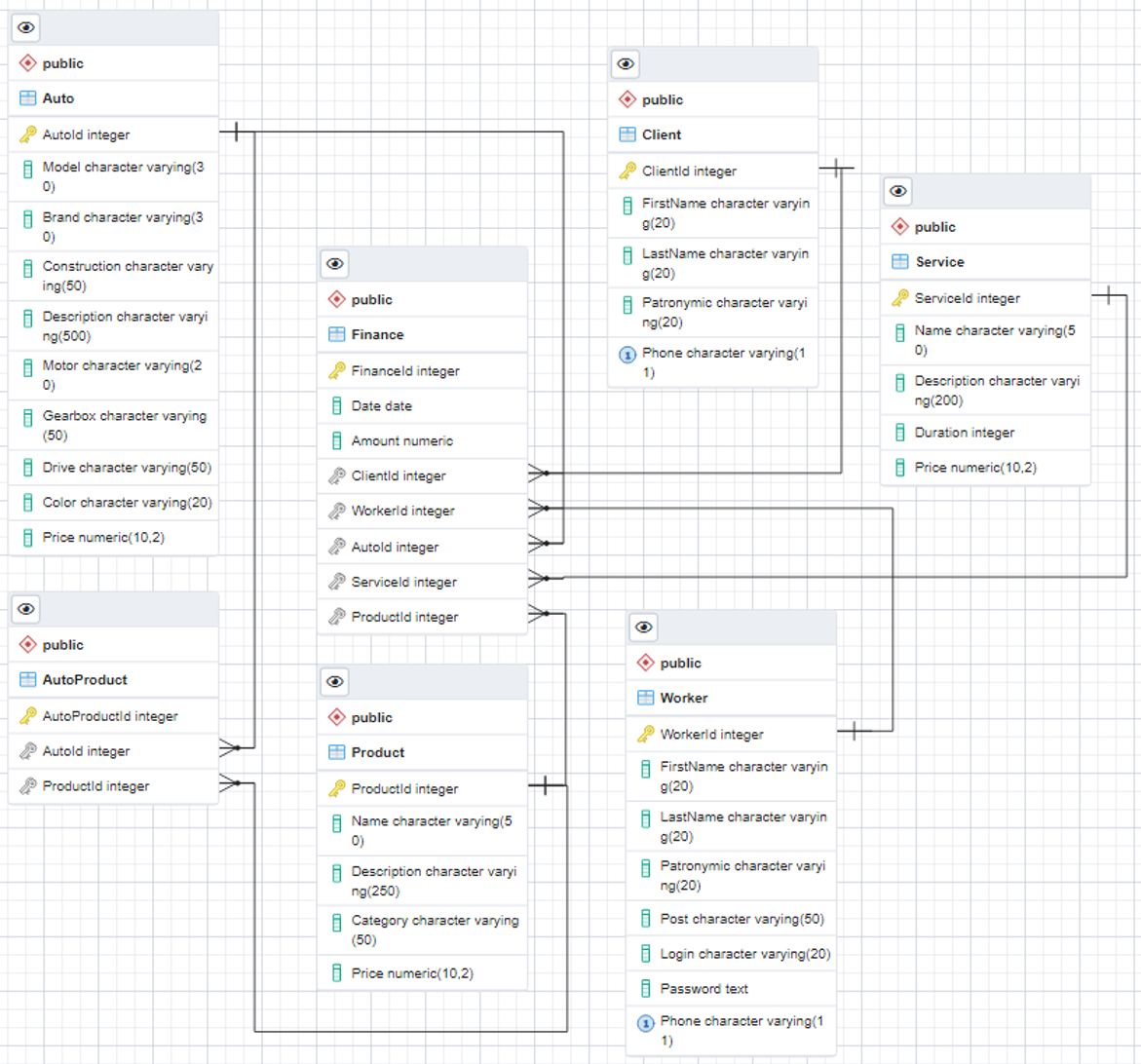
В целом, выбранные технические и программные средства соответствуют требованиям проекта и обеспечивают надежную и эффективную разработку базы данных и приложения для дилерского центра.

## Физическая структура

Физическая модель базы данных в контексте дилерского автоцентра описывает способ, которым данные хранятся в базе данных, а также их физическое представление в памяти компьютера. Важно понимать, что физическая модель содержит все детали, необходимые для создания и управления базой данных с помощью системы управления базами данных (СУБД).

В дилерском автоцентре физическая модель базы данных определяет, как автомобильная информация будет храниться в памяти компьютера. Например, она определит, какие таблицы будут созданы для хранения информации о марках автомобилей, клиентах, продажах и т.д. Она также определит, какие столбцы будут присутствовать в каждой таблице и какие типы данных будут использоваться для хранения конкретной информации, такой как название марки автомобиля, год выпуска или цена.

Физическая структура базы данных дилерского центра «АвтоМоська» представлена на рисунке 3.2.1

Рисунок 3.2.1 Физическая структура базы данных

В PostgreSQL, отсутствует прямая поддержка отношений многие ко многим между таблицами. Вместо этого используется подход с промежуточными таблицами для моделирования таких отношений.

Для связи таблиц «Auto» и «Product» через отношение многие ко многим создаём дополнительную таблицу «AutoProduct». Эта таблица будет служить связующим звеном между автомобилями и продуктами, позволяя установить связь между ними.

Таблица «AutoProduct» будет содержать столбцы, отражающие связь между «Auto» и «Product». В ней будут столбцы «auto\_id» и «product\_id», которые будут внешними ключами, указывающими на соответствующие записи в таблицах «Auto» и «Product». Таким образом, каждая запись в таблице «AutoProduct» будет представлять конкретную связь между определенным автомобилем и продуктом.

При удалении записей из таблиц «Auto» и «Product» используется опция «ON DELETE CASCADE». Таким образом, при удалении автомобиля или продукта из соответствующих таблиц, все связи, связанные с этими записями в таблице «AutoProduct», будут удалены. Это обеспечивает целостность данных и поддерживает согласованность между таблицами «Auto», «Product» и «AutoProduct» в базе данных.

При удалении записей из таблиц «Worker», «Client», «Auto», «Product» используется опция «ON DELETE SET NULL», соответствующие ссылки в таблице «Finance» будут установлены в значение NULL. Использование опции «ON DELETE SET NULL» позволяет обеспечить целостность данных в таблице «Finance» при удалении связанных записей из других таблиц.

Примеры запросов в СУБД PostgreSQL:

* Выборка категории из таблицы продуктов

SELECT "Name", "Category", "Price" FROM "Product" WHERE "Category" = 'Запчасти';

Данным запросом создаётся выборка данных из таблицы продуктов, где берётся категория является «Запчасти». Выборка осуществляется с помощью условного оператора «Where». Данный запрос нужен для поиска определённого наименования, например, для поиска его цены.

SELECT "FirstName", "LastName", "Post", "Phone" FROM "Worker" WHERE "Phone" LIKE '8909%';

Данным запросом создаётся выборка данных из таблицы работников, где номер телефона начинается с «8909». Выборка осуществляется с помощью условного оператора «Where» и оператора выборки «LIKE». Данный запрос нужен для поиска работников, должности, номера телефона, который начинается с «8909».

SELECT "Product\_Name", "Product\_Cost" FROM "Product" ORDER BY "Product\_Cost";

Данным запросом создаётся выборка продуктов и их цен из таблицы продуктов по порядку возрастания. Выборка осуществляется с помощью оператора выборки «ORDER BY». Данный запрос нужен для сортировки продукции и их цен по возрастанию.

SELECT O."Model", O."Brand", M."Amount", M."Date" FROM "Finance" AS M INNER JOIN "Auto" AS O ON M."AutoId" = O."AutoId"

Данным запросом создаётся выборка автомобилей, даты, цен из таблиц финансов и автомобилей. Выборка осуществляется с помощью условного оператора «Where» и оператора объединения «Inner Join». Данный запрос нужен для вывода автомобилей по финансовым операциям.

Заключение

В результате выполнения курсового проекта, связанного с разработкой базы данных и системы управления для дилерского автоцентра «АвтоМоська», была создана эффективная информационная система, способствующая улучшению бизнес-процессов, повышению конкурентоспособности и обеспечению удобного доступа к данным.

Проект является актуальным и востребованным, учитывая современные потребности дилерских компаний в автоматизированных информационных системах. Он позволяет эффективно управлять данными о клиентах, автомобилях, услугах, товарах и финансовых операциях, а также улучшить взаимодействие между отделами компании.

Реализованный функционал системы включает уровни доступа для разделения полномочий в управлении приложением. Рядовой персонал имеет возможность просматривать основную информацию о товарах, услугах.

В курсовом проекте успешно выполнены поставленные задачи, включающие изучение предметной области, определение данных для хранения, разработку логической структуры базы данных, выбор технических и программных средств, а также создание технического проекта.

База данных и система управления разработаны с использованием СУБД PostgreSQL и WPF-приложения, обеспечивающего динамическое управление данными. Это позволяет эффективно хранить, обрабатывать и анализировать информацию, а также управлять доступом к ней.

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. Джуба, С. Изучаем PostgreSQL 10 / С. Джуба, А. Волков. — Москва : ДМК Пресс, 2019. — 400 с. — ISBN 978-5-97060-643-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/116125;

3. Грошев А.С. Основы работы с базами данных [Электронный ресурс]. – Режим доступа: https://studfile.net/preview;

4. Карпова Т.С. Базы данных: модели, разработка, реализация. [Электронный ресурс]. – Режим доступа: http://www.intuit.ru/department/database;

5. Кузнецов С.Д. Введение в реляционные базы данных [Электронный ресурс]. – Режим доступа: https://intuit.ru/studies/courses/74/74/lecture/27919;

7. Документация PostgreSQL [Электронный ресурс]. – Режим доступа: https://postgrespro.ru/docs/postgresql.