

Министерство образования и науки высшего образования Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)

Факультет ИТР
Кафедра ПИН

КУРСОВАЯ
РАБОТА

по курсу Системы управления базами данных

на тему: АИС «Продажа б\у сотовых телефонов и аксессуаров»

Руководитель

к. т. н., доц. каф. ПИН

(уч. степень, звание)

Колпаков А.А

(фамилия, инициалы)

(подпись)

(дата)

Члены комиссии

Студент ПИН-123

(группа)

(подпись)

(Ф.И.О.)

Комиссаров Д.С.

(фамилия, инициалы)

(подпись)

(Ф.И.О.)

(подпись)

(дата)

Муром 2025

В данной курсовой работе разработана автоматизированная информационная система (АИС) для магазина по продаже и ремонту мобильных телефонов. Система позволяет вести учет б/у техники, аксессуаров и запчастей, оформлять операции скупки и продажи, а также контролировать процесс сервисного обслуживания. В качестве СУБД использована PostgreSQL, клиентская часть реализована на языке C#. Реализованы механизмы автоматического расчета себестоимости через триггеры и генерация аналитической отчетности.

This term paper describes the development of an automated information system (AIS) for a mobile phone sales and repair shop. The system enables inventory management of used devices, accessories, and spare parts, processing buyback and sales operations, and monitoring the repair workflow. PostgreSQL is used as the DBMS, and the client application is implemented in C#. Automation features include trigger-based cost calculation and analytical reporting.

Содержание

Введение.....	5
1 Анализ технического задания	6
Основные сущности предметной области:	6
Основные бизнес-процессы:	6
2 Разработка моделей данных	8
2.1 Структура связей.....	8
2.2 Описание концептуальной модели данных	9
2.3 Логическая модель данных	11
2.4 Физическая модель данных.....	13
2.5 Реализация активной логики БД (Триггеры).....	16
3 Разработка и реализация АИС	18
3.1 Реализация интерфейса и логики	18
3.2 Реализация триггеров и процедур	22
3.3 Руководство программиста	23
3.4 Руководство пользователя.....	23
4 Тестирование	25
Заключение	29
Список литературы	30
Приложение 1 Модели данных	31
Приложение 2 Текст программы	33
Приложение 3 Скриншоты программы	35

Изм.	Лист	№ докум.	Подп.	Дата
Разраб.	Комиссаров Д.С.			
Пров.	Колпаков А.А.			
Н. контр.				
Утв.				

МИВУ 09.03.04.07 ПЗ

Лит.	Лист	Листов
У	5	42

СУБД Продажа Б/У сотовых телефонов
и аксессуаров
Пояснительная записка

МИ ВлГУ
ПИН-123

Введение

Современные торговые предприятия всё чаще используют автоматизированные информационные системы для повышения эффективности своей деятельности. Магазины по продаже мобильных телефонов и аксессуаров не являются исключением: обработка больших объёмов информации о товарах, услугах и клиентах требует применения специализированных программных решений.

Разработка информационной системы для магазина, специализирующегося на продаже телефонов бывших в употреблении, новых аксессуаров и оказании услуг по обслуживанию устройств, позволит оптимизировать хранение и обработку данных, а также автоматизировать формирование отчётности. Внедрение такой системы обеспечит:

- снижение трудоёмкости ведения учёта;
- повышение точности и оперативности получения информации;
- возможность быстрого анализа финансовых показателей;
- улучшение качества обслуживания клиентов.

В рамках данной курсовой работы требуется спроектировать и реализовать базу данных, которая будет содержать сведения о товарах (телефонах, аксессуарах), услугах и операциях по продаже. Кроме того, необходимо предусмотреть хранение изображений, реализацию хранимых процедур и/или триггеров, а также возможность получения аналитических отчётов (по выручке, прибыли, количеству проданных товаров и оказанных услуг).

Таким образом, целью работы является создание информационной системы магазина продажи телефонов, которая обеспечит автоматизацию учёта, обработку данных и формирование аналитических отчётов на основе хранимой информации.

Изм.	Лист	№ докум.	Подп.	Дата

1 Анализ технического задания

Магазин, для которого создаётся система, осуществляет три основных вида деятельности:

- продажа телефонов бывших в употреблении.
- продажа новых аксессуаров (чехлы, защитные стёкла, зарядные устройства и др.).
- оказание сервисных услуг (наклейка защитной пленки, установка аксессуаров и т. п.).

Основные сущности предметной области:

- телефон: марка, модель, характеристики (память, цвет, состояние), цена приёма и продажи, фото.
- аксессуар: категория (чехол, пленка и др.), модель телефона, цена, фото.
- услуга: наименование, категория, стоимость.
- продажа: дата, клиент, проданный товар или оказанная услуга, количество, сумма.
- клиент: ФИО, контактные данные (опционально).
- сотрудник: может фиксироваться как исполнитель продажи или услуги.

Основные бизнес-процессы:

- учёт поступления товара – регистрация новых телефонов и аксессуаров, фиксация их характеристик.
- продажа товара или услуги – оформление операции продажи, расчёт суммы.
- формирование отчётности – получение данных о количестве проданных телефонов и аксессуаров, оказанных услугах, суммарной выручке и прибыли за выбранный период.

Телефоны поступают в магазин по цене приёма, которая отличается от цены продажи; разница образует прибыль.

Услуги и аксессуары имеют фиксированную цену.

Для анализа работы магазина необходимы отчёты по категориям товаров и услуг, а также по диапазону дат.

Изм.	Лист	№ докум.	Подп.	Дата

Для удобства клиентов и сотрудников требуется хранение изображений (например, фотографий телефонов и аксессуаров).

Контроль целостности и корректности данных должен обеспечиваться средствами СУБД (ограничения, триггеры, хранимые процедуры).

Таким образом, предметная область включает как торговые операции (товары, продажи), так и сервисные услуги. Для её корректного описания необходимо построить концептуальную, логическую и физическую модели данных, что станет основой для реализации автоматизированной информационной системы.

Изм.	Лист	№ докум.	Подп.	Дата

2 Разработка моделей данных

2.1 Структура связей

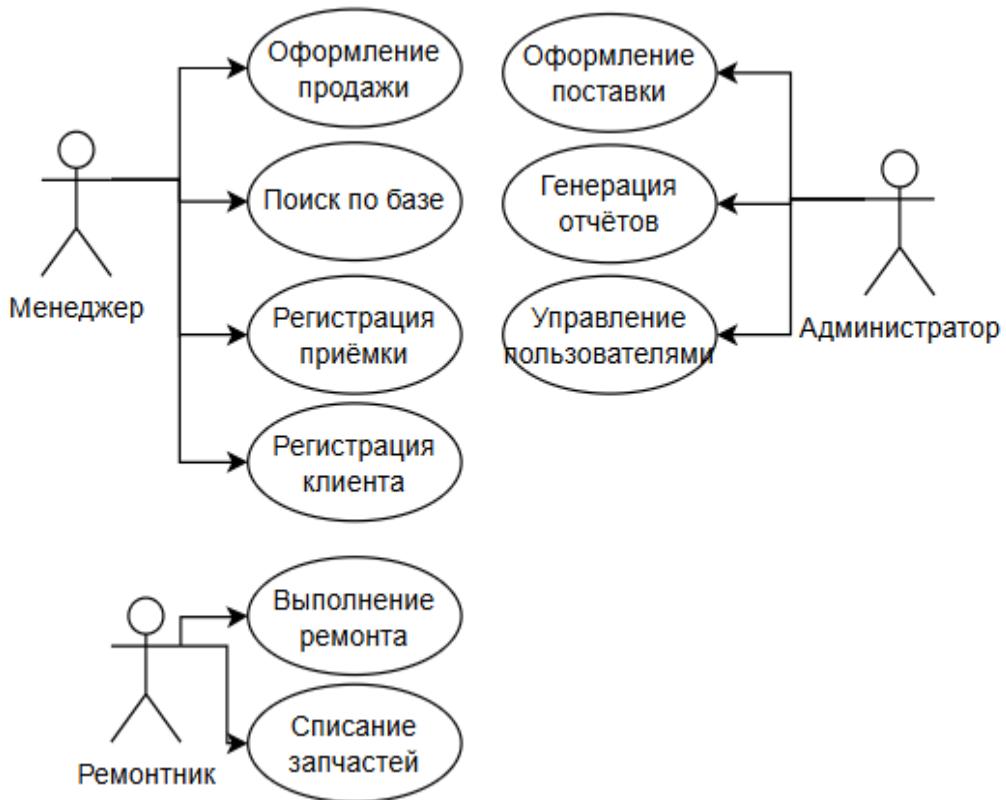


Рисунок 1 – UML Диаграмма “Магазин Б/У сотовых телефонов и аксессуаров”

Менеджер

Этот человек делает всё, что связано с деньгами и клиентами.

"Оформление продажи": Создает чек, продает телефон или аксессуар.

"Регистрация приемки": Оценивает телефон клиента и заносит его в базу (складка).

"Поиск по базе": Смотрит, есть ли чехол на iPhone 13 или готов ли телефон из ремонта.

"Регистрация клиента": Добавляет ФИО и телефон в базу.

Ремонтник (Мастер)

Этот человек работает только с техникой.

"Выполнение ремонта": Изменяет статус телефона (В ремонте → Готов).

Изм.	Лист	№ докум.	Подп.	Дата

"Списание запчастей": Указывает, что потратил экран или батарейку.

Администратор

Главный над всеми.

"Оформление поставки": Заносит приход товара от поставщиков.

"Генерация отчетов": Прибыль, зарплата, остатки.

"Управление пользователями": Создает логины для Менеджера и Мастера.

2.2 Описание концептуальной модели данных

Разработанная модель данных описывает информационную систему магазина по скупке, восстановлению и продаже мобильной техники. Модель включает следующие основные функциональные блоки:

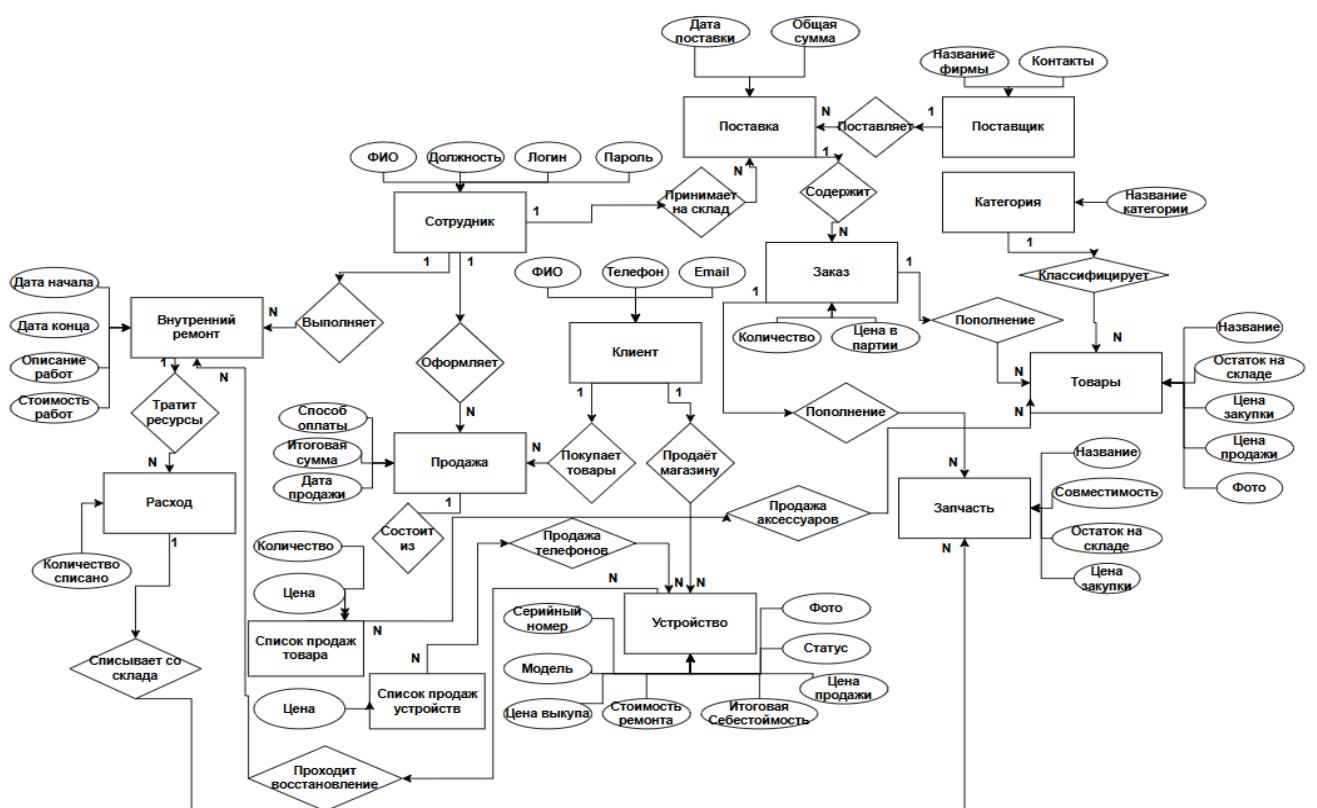


Рисунок 2 – Концептуальная модель данных

Ключевые роли

Взаимодействие в системе строится вокруг четырёх ключевых участников:

Изм.	Лист	№ докум.	Подп.	Дата	МИВУ 09.03.04 ПЗ	Лист 9

Сотрудник: Центральная сущность. Сотрудники имеют ролевой доступ. Один сотрудник может выполнять множество действий: принимать поставки, выполнять внутренние ремонты и оформлять продажи.

Клиент: В системе выполняет двойную функцию. Связь «Продает магазину» отражает процесс скупки б/у техники у населения, а связь «Покупает товары» — процесс приобретения восстановленных устройств или аксессуаров.

Ремонтник: В системе выполняет функцию ремонт. Он принимает телефоны, поломаны, добавляет запчасти со склада, пишет отчёты по ремонту телефонов.

Администратор: В системе выполняет все функции по программе. Он может просматривать среды работы обоих работников и взаимодействовать с ними, принимать на работу новых сотрудников, добавлять товары, категории товаров, заказывать товары и запчасти для починки, просматривать отчёт прибыли.

Складской учет и Поставки

Для разделения товарных потоков реализованы две отдельные сущности:

Товары: «Аксессуары» и сопутствующие товары для прямой перепродажи. Имеют «Цену закупки» и «Цену продажи». Классифицируются через справочник «Категория».

Запчасти: Комплектующие для внутреннего использования. Имеют атрибут «Совместимость» и не предназначены для прямой продажи клиенту.

Процесс пополнения склада:

Сущность «Поставка» фиксирует факт прихода груза (дата, общая сумма).

Сущность «Заказ» детализирует, какие именно Товары или Запчасти пришли, в каком «Количестве» и по какой «Цене в партии».

Учет устройств и Восстановление

Центральным объектом модели является «Устройство»:

Это уникальный товар.

«Устройство» попадает в систему от «Клиента», при этом фиксируется «Цена выкупа».

Поле «Итоговая себестоимость» накапливает все затраты.

Процесс восстановления:

Изм.	Лист	№ докум.	Подп.	Дата	Лист	10
					МИВУ 09.03.04 ПЗ	

Внутренний ремонт: Связывает «Устройство» и «Сотрудника». Содержит «Описание работ» и временные метки.

Расход: Сущность, отражающая списание материалов. Она связывает ремонт с конкретной «Запчастью», уменьшая её остаток на складе.

Продажи

Реализация товаров клиентам организована через блок продаж:

Продажа: Фиксирует дату, итоговую сумму и сотрудника, оформившего сделку.

Список продаж товара: Оформляет количество сколько одинакового товара куплено и его цену.

Список продаж устройств: Оформляет цену на продажу устройства.

Позиция может ссылаться на «Товары».

Или позиция может ссылаться на уникальное Устройство.

2.3 Логическая модель данных

Логическая модель данных АИС «Продажа сотовых телефонов» построена на принципах реляционной архитектуры. Основная задача модели — обеспечить прослеживаемость жизненного цикла каждого уникального устройства (по IMEI) и количественный учет аксессуаров и запчастей.

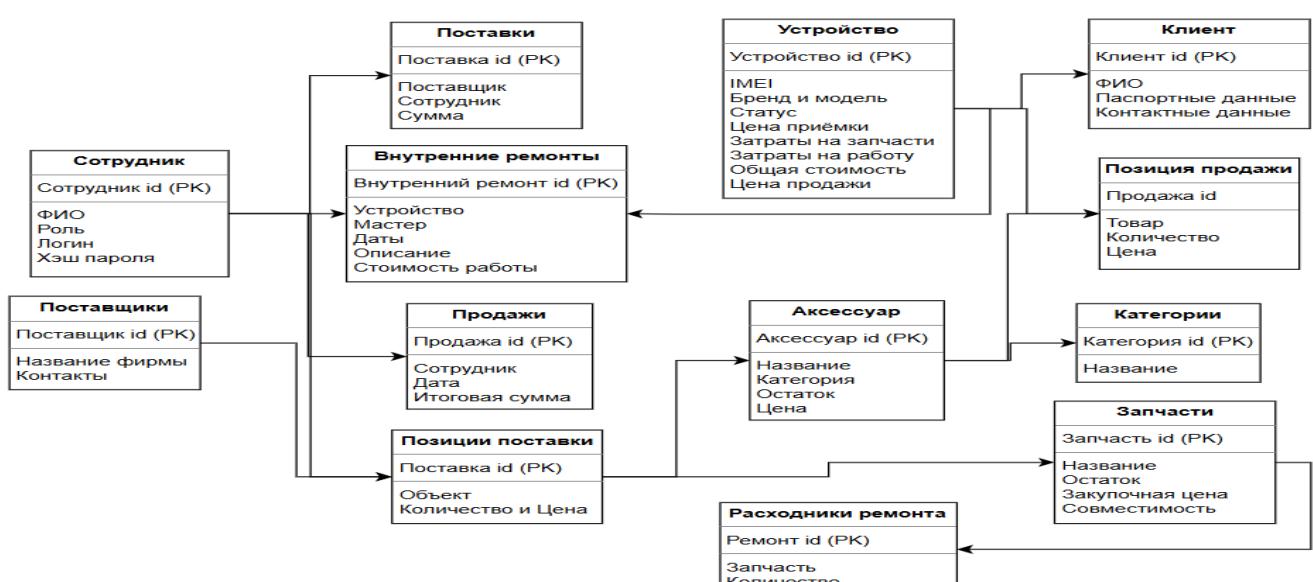


Рисунок 3 – Логическая модель данных

Изм.	Лист	№ докум.	Подп.	Дата

- контур субъектов: хранение данных о персонале (Employees), клиентах (Clients) и поставщиках (Suppliers).
- контур складского учета: классификация (Categories) и количественный учет товаров (Accessories) и деталей (Parts).
- контур уникального учета (Devices): детальное описание каждого мобильного аппарата, его технических характеристик, истории владения и накопленной себестоимости.
- контур транзакций: регистрация операций поставки (Supplies), сервисного обслуживания (Internal_Repairs) и реализации (Sales).

Ниже приведено детальное описание назначения каждой сущности:

Сотрудники (Employees): Обеспечивает безопасность системы через разграничение прав доступа (Администратор, Менеджер, Мастер). Поля Login и Password_hash используются для аутентификации.

Клиенты (Clients): Содержит персональные данные, необходимые для юридического оформления договоров скупки б/у техники.

Категории (Categories): Служит иерархическим справочником для группировки аксессуаров и услуг.

Аксессуары (Accessories): Описывает товары массового потребления. Ключевые поля Stock_quantity и Sale_price позволяют контролировать остатки и розничную выручку.

Запчасти (Parts): Учитывает компоненты, необходимые для восстановления техники. Атрибут Compatibility позволяет мастеру быстро подбирать детали к конкретным моделям.

Устройства (Devices): Центральный объект системы. Использует поле Total_cost (автоматически вычисляемое на основе закупки и затрат на ремонт) для определения минимально допустимой цены продажи.

Ремонты (Internal_Repairs): Фиксирует процесс восстановления устройства. Связывает аппарат с конкретным мастером и хранит технический отчет о работе.

Изм.	Лист	№ докум.	Подп.	Дата

Расходники (Repair_Consumables): Промежуточная сущность, реализующая связь «многие-ко-многим» между ремонтом и запчастями, позволяя списывать несколько деталей на один аппарат.

Продажи (Sales) и Позиции продажи (Sale_Items): Реализуют структуру электронного чека. Позволяют в рамках одной операции реализовывать как уникальные телефоны, так и аксессуары.

Поставки (Supplies), Поставщики (Suppliers) и Позиции поставки (Supply_Items): Обеспечивают автоматизацию прихода ТМЦ на склад, фиксируя закупочные цены и ответственных лиц.

2.4 Физическая модель данных

В качестве Системы Управления Базами Данных (СУБД) для реализации проекта выбрана PostgreSQL. Выбор обусловлен поддержкой строгой типизации, механизмами обеспечения целостности данных, наличием типов данных для финансовых вычислений (DECIMAL) и возможностью использования вычисляемых столбцов.

База данных состоит из 12 взаимосвязанных таблиц. Для именования сущностей использован стиль *snake_case* на английском языке.

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

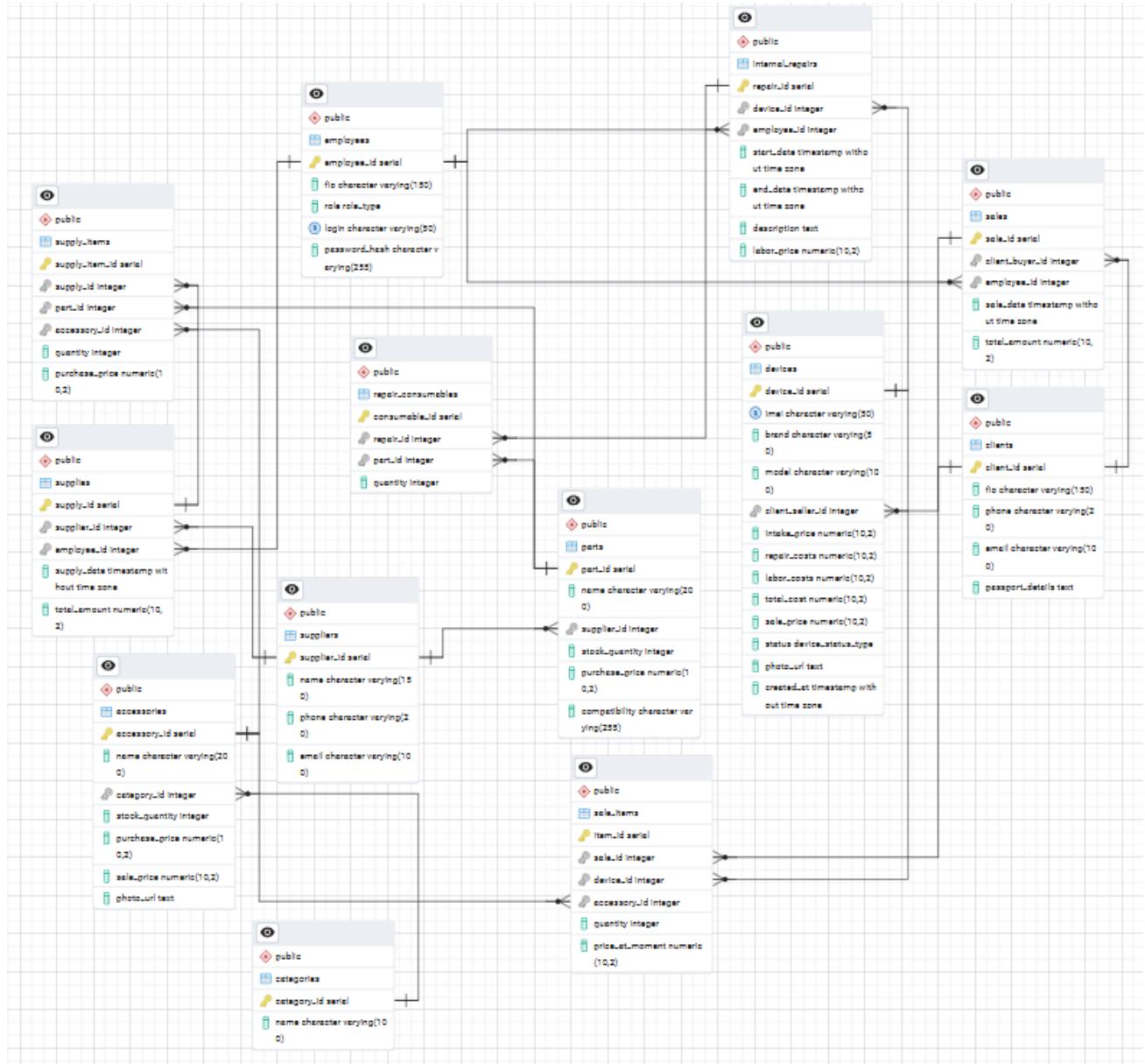


Рисунок 3 – Физическая модель данных

Справочные таблицы (Dictionaries)

“categories” Классификатор товаров. Содержит поля category_id (PK, SERIAL) и name (VARCHAR).

“Suppliers” Список контрагентов. Хранит supplier_id (PK), название, телефон и email.

“clients” База данных физических лиц (продавцов б/у техники и покупателей). Содержит паспортные данные (для скупки) и контакты.

Изм.	Лист	№ докум.	Подп.	Дата

“Employees” Пользователи системы. Поле role реализовано через тип ENUM ('admin', 'manager', 'master'), что ограничивает права доступа на уровне БД. Пароли хранятся в виде хеша (password_hash).

Товарно-материальные ценности (Inventory)

Реализовано физическое разделение товаров на две таблицы для исключения пересортицы:

Таблица “parts” (Запчасти)

Используются только для ремонтов.

“part_id”: Первичный ключ.

“purchase_price” (DECIMAL(10,2)): Цена закупки (себестоимость).

“stock_quantity”: Текущий остаток. Ограничение CHECK (stock_quantity >= 0) запрещает отрицательные остатки.

“Compatibility”: Текстовое поле для указания совместимых моделей.

Таблица “accessories” (Аксессуары/Товары)

Товары для прямой продажи.

“accessory_id”: Первичный ключ.

“sale_price” (DECIMAL(10,2)): Розничная цена.

“category_id”: Внешний ключ на таблицу категорий

Учет устройств (devices)

Центральная таблица системы, хранящая уникальные единицы техники.

“imei”: Уникальный идентификатор (VARCHAR(50) UNIQUE).

“Status”: Поле типа ENUM ('in_stock', 'on_repair', 'ready_for_sale', 'sold').

Финансовые поля:

“intake_price”: Цена выкупа у клиента.

“repair_costs”: Накопительная стоимость запчастей.

“labor_costs”: Стоимость работ мастера.

“total_cost”: Вычисляемый столбец (GENERATED ALWAYS AS ... STORED).

СУБД автоматически суммирует три предыдущих поля, гарантируя актуальность полной себестоимости без лишних запросов.

Изм.	Лист	№ докум.	Подп.	Дата

Операционные таблицы (Transactions)

Таблицы “supplies” и “supply_items” (Поставки)

Реализуют связь «Многие-ко-Многим» между Поставщиком и Товарами.

В “supply_items” используется механизм Exclusive Foreign Keys через ограничение CHECK: в одной строке может быть заполнен либо “part_id”, либо “accessory_id”, но не оба сразу.

Таблицы “internal_repairs” и “repair_consumables” (Ремонт)

“internal_repairs” связывает устройство и сотрудника-мастера.

“repair_consumables” фиксирует списание запчастей. Внешний ключ настроен с каскадным удалением (ON DELETE CASCADE).

Таблицы “sales” и “sale_items” (Продажи)

“Sales”: Заголовок чека с итоговой суммой.

“sale_items”: Позиции чека. Реализована логика: если заполнен “device_id” (продажа телефона), то количество (quantity) принудительно равно 1.

2.5 Реализация активной логики БД (Триггеры)

Для обеспечения целостности данных и автоматизации бизнес-процессов на физическом уровне разработаны три PL/pgSQL функции и соответствующие триггеры.

Автоматизация складского учета (trg_supply_stock)

Событие: Добавление записи в таблицу supply_items (состав поставки).

Логика: Триггер определяет тип товара (запчасть или аксессуар) и автоматически увеличивает значение поля stock_quantity в соответствующей таблице (parts или accessories). Также обновляется закупочная цена на актуальную.

Расчет стоимости ремонта (trg_repair_parts)

Событие: Добавление записи в таблицу repair_consumables (списание запчасти мастером).

Логика:

Уменьшает остаток запчасти в таблице “parts”.

Изм.	Лист	№ докум.	Подп.	Дата

Берет стоимость запчасти и прибавляет её к полю “repair_costs” в таблице “devices”.

Благодаря вычисляемому столбцу “total_cost”, итоговая себестоимость телефона пересчитывается мгновенно.

Обработка продаж (trg_sale_processing)

Событие: Добавление записи в “sale_items”.

Логика:

При продаже аксессуара: уменьшает его остаток на складе.

При продаже телефона: автоматически меняет его статус в таблице “devices” на 'sold' (продан), блокируя возможность повторной продажи.

Ограничения целостности (Constraints)

На физическом уровне реализованы следующие правила:

PRIMARY KEY: Гарантирует уникальность каждой записи.

FOREIGN KEY: Обеспечивает ссылочную целостность (нельзя продать товар несуществующего клиента).

CHECK (quantity > 0): Запрещает ввод отрицательного или нулевого количества товаров.

CHECK (X IS NOT NULL OR Y IS NOT NULL): Гарантирует, что в позиции чека или накладной обязательно указан товар.

UNIQUE (imei): Запрещает дублирование телефонов с одинаковым серийным номером.

Изм.	Лист	№ докум.	Подп.	Дата

3 Разработка и реализация АИС

Автоматизированная информационная система (АИС) для магазина продажи сотовых телефонов реализована в виде десктопного приложения на языке программирования C# с использованием библиотеки WinForms для создания графического интерфейса и Npgsql для взаимодействия с базой данных PostgreSQL.

Приложение состоит из нескольких основных форм, каждая из которых отвечает за конкретные бизнес-процессы: авторизацию, продажи, ремонт, административное управление, складу устройств и генерацию отчетов.

3.1 Реализация интерфейса и логики

Интерфейс приложения разработан с учетом удобства пользователей: формы содержат элементы управления, такие как кнопки, списки, таблицы DataGridView и комбо-боксы для выбора данных. Логика работы форм тесно интегрирована с базой данных через SQL-запросы, выполняемые с помощью NpgsqlCommand и NpgsqlDataAdapter. Ниже приведено подробное описание каждой формы и ее функционала.

Форма авторизации (LoginForm)

Форма авторизации работает входной точкой в систему и делает проверку учетных данных пользователя. При запуске приложения отображается интерфейс с полями для ввода логина и пароля (txtLogin и txtPassword).

Основные функции:

- хэширование пароля: Введенный пароль преобразуется в хэш с использованием алгоритма SHA256 для безопасного сравнения с хранимым в БД значением (функция HashPassword).

- запрос к БД: Выполняется SQL-запрос SELECT для проверки логина и хеша пароля в таблице employees. Если данные верны, извлекается роль пользователя (role) и ФИО (fio).

- переход по роли: В зависимости от роли открывается соответствующая форма: ManagerForm для роли "manager", MasterForm для "master", AdminForm для

Изм.	Лист	№ докум.	Подп.	Дата

"admin". Форма LoginForm скрывается (this.Hide()), а после закрытия основной формы приложение завершается.

- обработка ошибок: При неверных данных выводится сообщение об ошибке.

Кнопка выхода (btnExit) завершает приложение.

Форма менеджера (ManagerForm)

Форма менеджера предназначена для работы с каталогами товаров и услуг, поиска, формирования корзины и оформления продаж. Интерфейс включает панель каталога (flowPanelCatalog), список корзины (lstCart), поле поиска (txtSearch) и кнопки для переключения режимов (устройства, аксессуары, услуги).

Основные функции:

- загрузка каталогов: Методы LoadDevices, LoadAccessories, LoadServices извлекают данные из БД с использованием SQL-запросов с JOIN (для категорий) и фильтром по статусу (ready_for_sale для устройств) или категории (исключая услуги для аксессуаров). Карточки товаров создаются динамически с помощью UserControl ProductCard, отображающего ID, название, цену и фото (если доступно).

- поиск: При изменении текста в txtSearch (событие TextChanged) каталог перезагружается с условием ILIKE для нечувствительного к регистру поиска.

- корзина: При клике на карточку товар добавляется в lstCart как объект CartItem. Сумма обновляется в lblTotalSum. Кнопка очистки (btnClearCart) сбрасывает корзину.

- оформление продажи: Кнопка btnPay проверяет корзину, создает транзакцию: вставляет запись в sales, затем в sale_items для каждого элемента, обновляет статус устройств или остатки аксессуаров. Если успешно, выводит сообщение и обновляет каталог.

- дополнительно: Кнопка скупки (btnOpenBuyback) открывает BuybackForm для регистрации новых устройств.

Изм.	Лист	№ докум.	Подп.	Дата

Форма мастера (MasterForm)

Форма мастера фокусируется на ремонте устройств. Интерфейс разделен на панели: очередь ремонтов (dgvRepairTasks), детали устройства (lblDeviceModel, txtProblemDesc, pbDevicePhoto), использованные запчасти (dgvUsedParts, cmbParts) и отчет о работе (txtWorkResult, numLaborPrice).

Основные функции:

- загрузка очереди: Метод LoadRepairTasks извлекает ремонты со статусом 'in_process' из таблицы internal_repairs с JOIN на devices. Отображает в DataGridView, скрывая технические поля.
- выбор ремонта: При клике на строку (CellClick) загружает детали: модель, проблему, фото. Загружает использованные запчасти (LoadUsedParts) с расчетом сумм.
- добавление запчастей: Кнопка btnAddPart проверяет наличие (stock_quantity > 0), списывает в транзакции и добавляет в repair_consumables. Обновляет таблицы.
- завершение ремонта: Кнопка btnFinish устанавливает фиксированную цену работы (5000 руб.), обновляет internal_repairs (статус 'ready', отчет) и devices (статус 'ready_for_sale', labor_costs, sale_price с наценкой). Использует транзакцию.
- отчеты: Кнопка btnOpenRepairReport открывает RepairsReportForm для просмотра завершенных ремонтов.
- обновление: Кнопка btnRefresh перезагружает очередь.

Форма администратора (AdminForm)

Форма администратора предоставляет инструменты для управления системой. Включает вкладки для категорий, поставщиков, сотрудников, поставок, склада и отчетов.

Основные функции:

- управление категориями: Загрузка в lstCategories, добавление (btnAddCategory) и удаление (btnDeleteCategory) с SQL-запросами.
- управление поставщиками: Отображение в dgvSuppliers, добавление через AddSupplierForm, удаление (btnDeleteSupplier).

Изм.	Лист	№ докум.	Подп.	Дата

- управление сотрудниками: Загрузка в dgvEmployees, добавление (btnAddUser) с хэшированием пароля и ENUM-ролью, удаление (btnDeleteUser).
- оформление поставок: Выбор поставщика и товаров (cmbSupplySupplier, cmbSupplyProduct), добавление в черновик (lstSupplyDraft), оформление (button2_Click) с транзакцией: вставка в supplies и supply_items, обновление остатков и цен.
- управление складом: Отображение в dgvStock (UNION accessories и parts), добавление фото (btnSetPhoto) с копированием в "Photos".
- отчеты: График выручки (chartRevenue) по дням, расчет прибыли (lblTotalProfit) с SQL-запросами по датам.
- восстановление: Кнопки btnBackup и btnRestore вызывают pg_dump/psql для создания/восстановления SQL-файлов.
- управление ролевыми формами: Кнопки для открытия ManagerForm и MasterForm.

Форма скупки (BuybackForm)

Форма для регистрации скупки устройств. Включает поля для клиента (ФИО, паспорт, телефон), устройства (модель, IMEI, цена, состояние, фото).

- загрузка фото: Кнопка btnUploadPhoto открывает диалог, копирует файл в "Photos".
- регистрация: Кнопка btnConfirmBuy проверяет поля, создает транзакцию: добавляет клиента в clients, устройство в devices (с статусом по состоянию: 'ready_for_sale' если отлично, иначе 'on_repair'), фото. Если ремонт - создает запись в internal_repairs.
- логика состояния: ComboBox cmbCondition определяет статус и цену продажи.

Форма отчётов по ремонтам (RepairsReportForm)

Форма для просмотра завершенных ремонтов. Включает даты (dtpStart, dtpEnd) и таблицу (dgvRepairsHistory).

Основные функции:

Изм.	Лист	№ докум.	Подп.	Дата

- загрузка данных: Кнопка btnShow извлекает завершенные ремонты ('ready') с JOIN на devices и employees, фильтром по датам.
- оформление таблицы: Метод DesignTable настраивает зебру, шрифты, перенос текста, ширину колонок для удобства чтения.

3.2 Реализация триггеров и процедур

Для автоматизации бизнес-логики на уровне БД реализованы триггеры на языке PL/pgSQL. Они обеспечивают целостность данных без дополнительного кода в приложении. Триггеры срабатывают на события INSERT в операционных таблицах.

- триггер trg_supply_stock (пополнение склада):

Событие: После INSERT в supply_items.

Логика: Определяет тип (part_id или accessory_id), увеличивает stock_quantity на quantity, обновляет purchase_price на price. Это автоматизирует учет поступлений, предотвращая ручные обновления.

- триггер trg_repair_parts (расчет ремонта):

Событие: После INSERT в repair_consumables.

Логика: Уменьшает stock_quantity запчасти в parts на quantity. Прибавляет (quantity * purchase_price) к repair_costs устройства. Вычисляемый столбец total_cost в devices автоматически пересчитывается (intake_price + repair_costs + labor_costs).

- триггер trg_sale_processing (обработка продаж):

Событие: После INSERT в sale_items.

Логика: Если accessory_id - уменьшает stock_quantity в accessories. Если device_id - устанавливает статус 'sold' в devices, блокируя повторную продажу.

В C#-коде транзакции (conn.BeginTransaction) используются для групп операций: продажи (вставка sales, sale_items, обновление остатков), поставки (supplies, supply_items, обновление stock), ремонт (обновление internal_repairs, devices, списание parts). Это гарантирует rollback при ошибках.

Изм.	Лист	№ докум.	Подп.	Дата

3.3 Руководство программиста

- формы: LoginForm (авторизация), ManagerForm (продажи), MasterForm (ремонт), AdminForm (управление), BuybackForm (складка), RepairsReportForm (отчеты), AddSupplierForm (поставщики).
- классы: Database (подключение), CartItem (элемент корзины), SupplyItemDraft (чертёж поставки), ProductCard (UserControl для карточек).
- ключевые методы: В формах - Load для загрузки данных, события Click для действий, транзакции в btnPay, btnFinish, button2_Click.
- триггеры: Определены в БД (PL/pgSQL), интегрированы автоматически при INSERT.

3.4 Руководство пользователя

Авторизация

- запустите приложение Megalight.exe.
- введите логин и пароль, нажмите "Войти"
- при ошибке - проверьте данные. Выход - кнопка "Выход".

Работа менеджера (ManagerForm)

- просмотр каталога: Выберите "Смартфоны", "Аксессуары" или "Услуги"
- поиск: Введите текст в поле поиска.
- добавление в корзину: Кликните на карточку товара.
- оформление: Нажмите "Оплатить" (обновит БД).
- складка: Кнопка "Открыть складку" - заполните форму, загрузите фото, подтвердите

Работа мастера (MasterForm)

- очередь ремонтов: Просмотрите таблицу
- выбор: Кликните строку - загружаются детали.
- добавление запчастей: Выберите из комбо, нажмите "+"
- завершение: Заполните отчет, нажмите "Завершить"
- отчеты: Кнопка "Отчёты" - укажите даты, нажмите "Показать"

Изм.	Лист	№ докум.	Подп.	Дата

Работа администратора (AdminForm)

- категории: Добавьте/удалите в списке.
- поставщики: Добавьте через форму, удалите в таблице.
- сотрудники: Добавьте (ФИО, логин, пароль, роль), удалите.
- поставки: Выберите поставщика, добавьте товары в черновик, оформите
- склад: Просмотрите таблицу, добавьте фото к товару.
- отчеты: Укажите даты, нажмите "Показать"
- восстановление: Выберите файл, создайте или восстановите, в зависимости от кнопки.

Изм.	Лист	№ докум.	Подп.	Дата

МИВУ 09.03.04 ПЗ

Лист

24

4 Тестирование

- авторизация и роли:

Проверена авторизация в LoginForm: ввод верных логина и хэша пароля приводит к открытию формы по роли (manager, master, admin). Неверные данные вызывают сообщение об ошибке. Тестируались все роли: администратор получает доступ к управлению, менеджер — к продажам, мастер — к ремонтам.

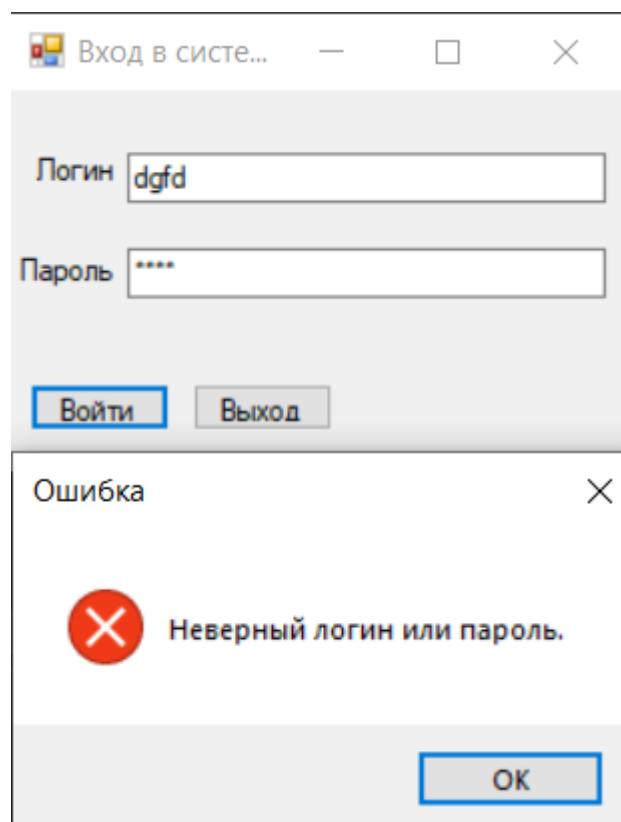


Рисунок 4 – Проверка входа

- продажи и корзина

Загрузка каталогов: Проверено отображение устройств (статус 'ready_for_sale'), аксессуаров и услуг с фото (из photo_url). Поиск по ILIKE работает без учета регистра, фильтруя результаты.

Изм.	Лист	№ докум.	Подп.	Дата

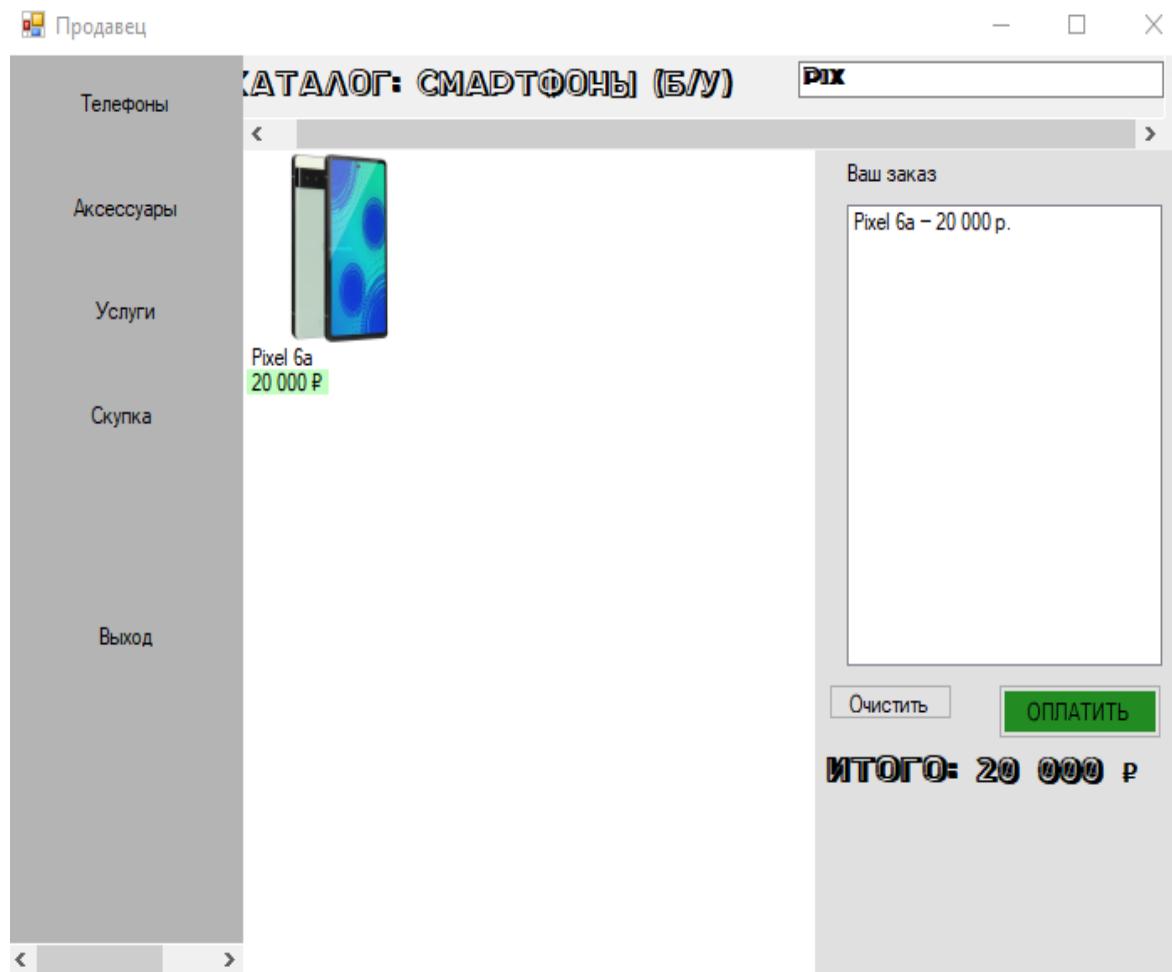


Рисунок 5 – Поиск и отображение фото товара и добавление в корзину

Добавление в корзину: Клик на ProductCard добавляет CartItem, обновляет сумму. Тестировалось добавление нескольких товаров, очистка корзины.

Оформление продажи: Транзакция вставляет записи в sales и sale_items, обновляет остатки (accessories) или статус ('sold' для devices). Проверено через триггер trg_sale_processing: повторная продажа устройства невозможна.

- скупка устройства

Регистрация: Заполнение полей клиента и устройства, загрузка фото (копирование в "Photos"). Если состояние "отлично" — статус 'ready_for_sale' с наценкой 30%; иначе — 'on_repair' с созданием записи в internal_repairs.

Проверено UNIQUE на IMEI: дубли блокируются. Краевой случай: Отсутствие фото — вставка DBNull.

Изм.	Лист	№ докум.	Подп.	Дата

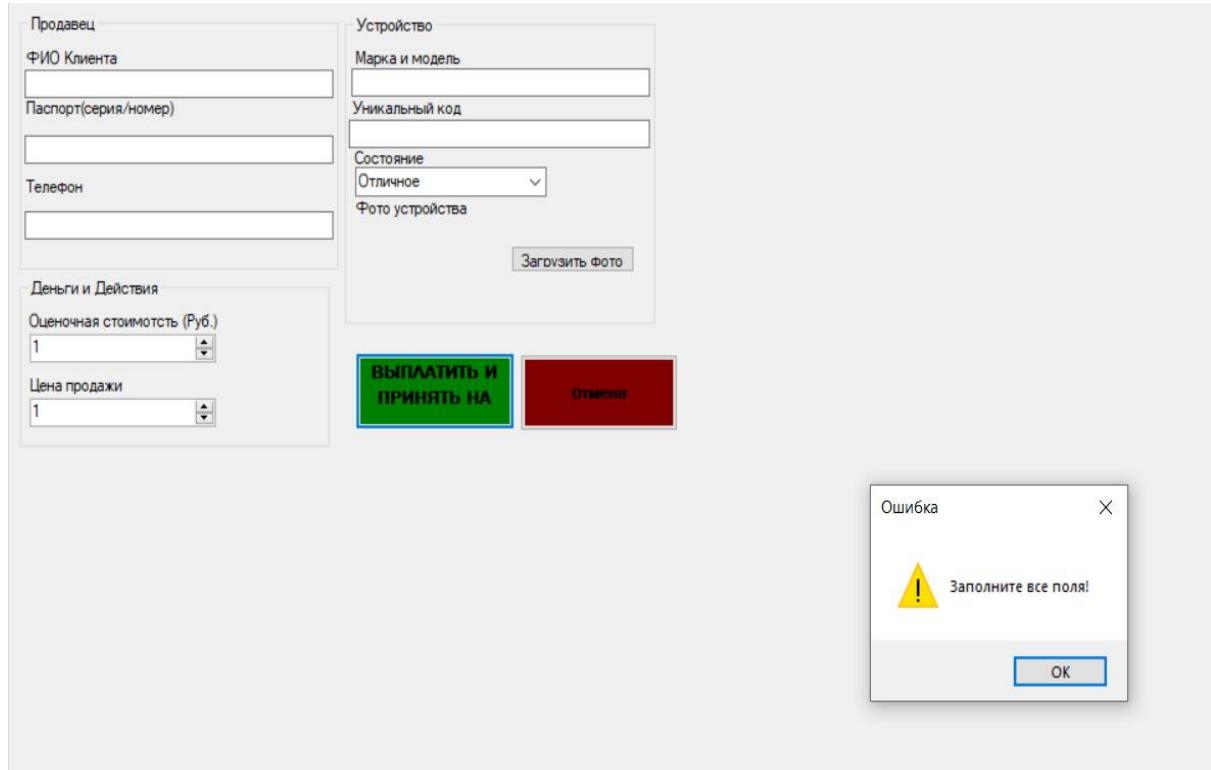


Рисунок 6 – Ошибка полей

- ремонт

Очередь: Загрузка ремонтов 'in_process' с JOIN. Выбор строки отображает детали, фото и использованные запчасти.

Добавление запчастей: Проверка наличия, списание в транзакции, обновление repair_consumables и repair_costs через trg_repair_parts.

Завершение: Установка фиксированной цены (5000 руб.), обновление статуса 'ready', расчет sale_price. Тестировалось накопление total_cost.

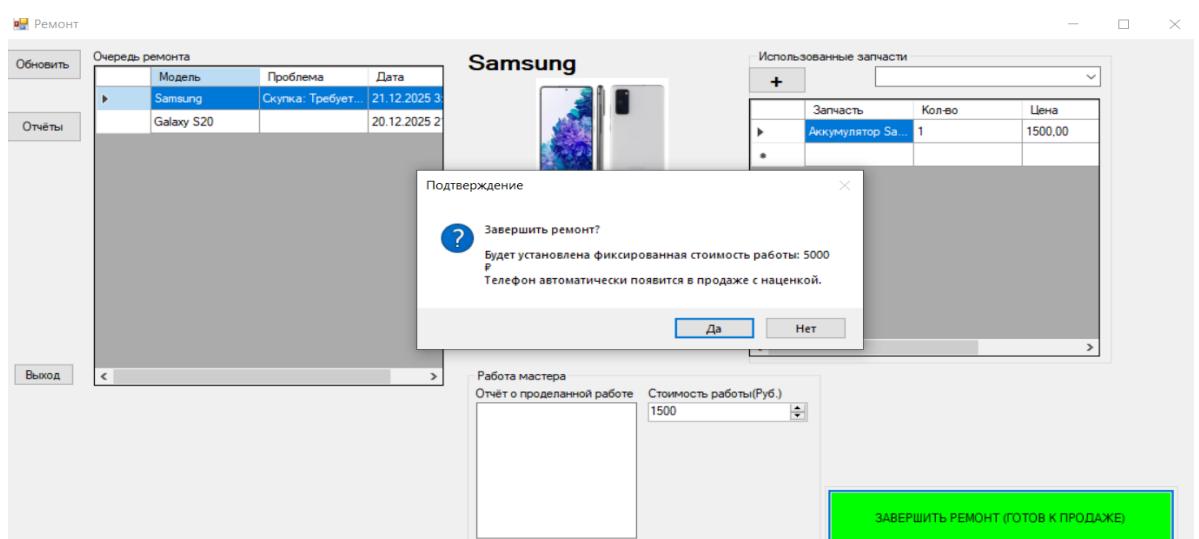


Рисунок 7 – Полная стоимость ремонта с наценкой

Иэм.	Лист	№ докум.	Подп.	Дата

- управление

Категории/поставщики/сотрудники: Добавление, удаление, хэширование паролей. Проверено ENUM для ролей.

Поставки: Черновик (SupplyItemDraft), транзакция для supplies/supply_items, пополнение stock через trg_supply_stock.

Склад: Отображение UNION, добавление фото.

Отчеты: График выручки, прибыль по датам.

- отчёты по ремонтам

Фильтр по датам: Загрузка завершенных ремонтов, форматирование DataGridView (перенос текста, зебра).

-триггеры

trg_supply_stock — пополнение stock при поставке (проверено увеличение quantity).

trg_repair_parts — списание и обновление repair_costs (тест на отрицательные остатки — блокируется).

trg_sale_processing — обновление статуса/остатков при продаже.

- ограничения

CHECK (stock_quantity >= 0) — предотвращает отрицательные значения.

UNIQUE (imei) — блокирует дубли.

FOREIGN KEY — невозможно удалить связанную запись.

- транзакции

Проверен rollback: симуляция ошибки в продаже — данные не изменяются.

Изм.	Лист	№ докум.	Подп.	Дата

Заключение

В ходе выполнения курсовой работы была разработана автоматизированная информационная система для магазина по продаже сотовых телефонов, специализирующегося на б/у устройствах, новых аксессуарах и услугах по обслуживанию.

Создана база данных с концептуальной, логической и физической моделями, включающими сущности (устройства, аксессуары, запчасти, продажи, ремонты), триггеры для автоматизации (trg_supply_stock, trg_repair_parts, trg_sale_processing) и хранение изображений в поле photo_url. Обеспечена целостность данных через ограничения (CHECK, UNIQUE, FOREIGN KEY) и транзакции в коде. Тестирование подтвердило надежность системы на тестовом наборе данных (>50 записей). Обеспечена целостность данных через ограничения (CHECK, UNIQUE, FOREIGN KEY) и транзакции в коде. Тестирование подтвердило надежность системы на тестовом наборе данных (>50 записей).

Разработанная АИС снижает трудоемкость учета, повышает оперативность и точность информации, позволяет анализировать финансовые показатели.

Изм.	Лист	№ докум.	Подп.	Дата

Список литературы

1. Троелсен, Э. Джепикс, Ф. Язык программирования C# 12 и платформа .NET 8. — 11-е изд. — М.: Вильямс, 2024
2. Албахари, Дж. С# 12. Справочник. Полное описание языка. — СПб.: Питер, 2024
3. Подбельский В. В. Разработка приложений на C# с использованием СУБД PostgreSQL. — М.: ДМК Пресс, 2020.
4. Моргунов, Е. П. PostgreSQL. Основы языка SQL: учебное пособие. — М.: Postgres Professional, 2023
5. Васильев, А. Н. SQL. Понятное руководство по управлению данными. — СПб.: Питер, 2024
6. Лодон, Дж. Информационные системы управления. — 17-е изд. — СПб.: Питер, 2024
7. Григорьев, М. Н. Логистика в розничной торговле: управление запасами. — М.: Юрайт, 2023

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

Приложение 1 Модели данных

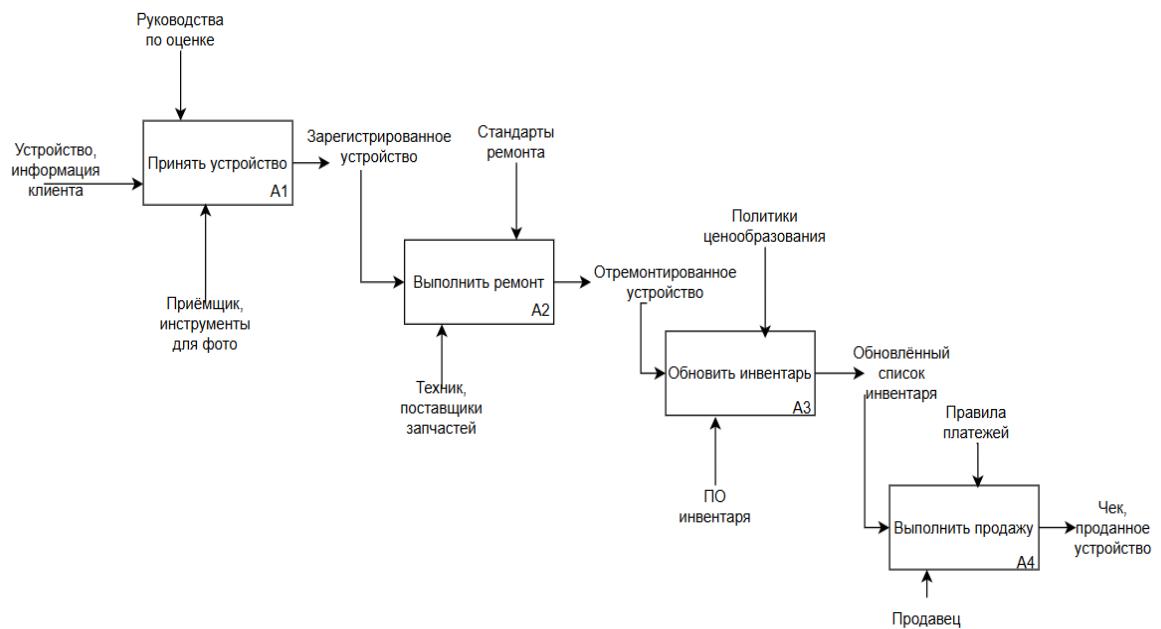


Рисунок 1.1 – Контекстная диаграмма «Магазин Б/У телефонов и аксессуаров»

Изм.	Лист	№ докум.	Подп.	Дата

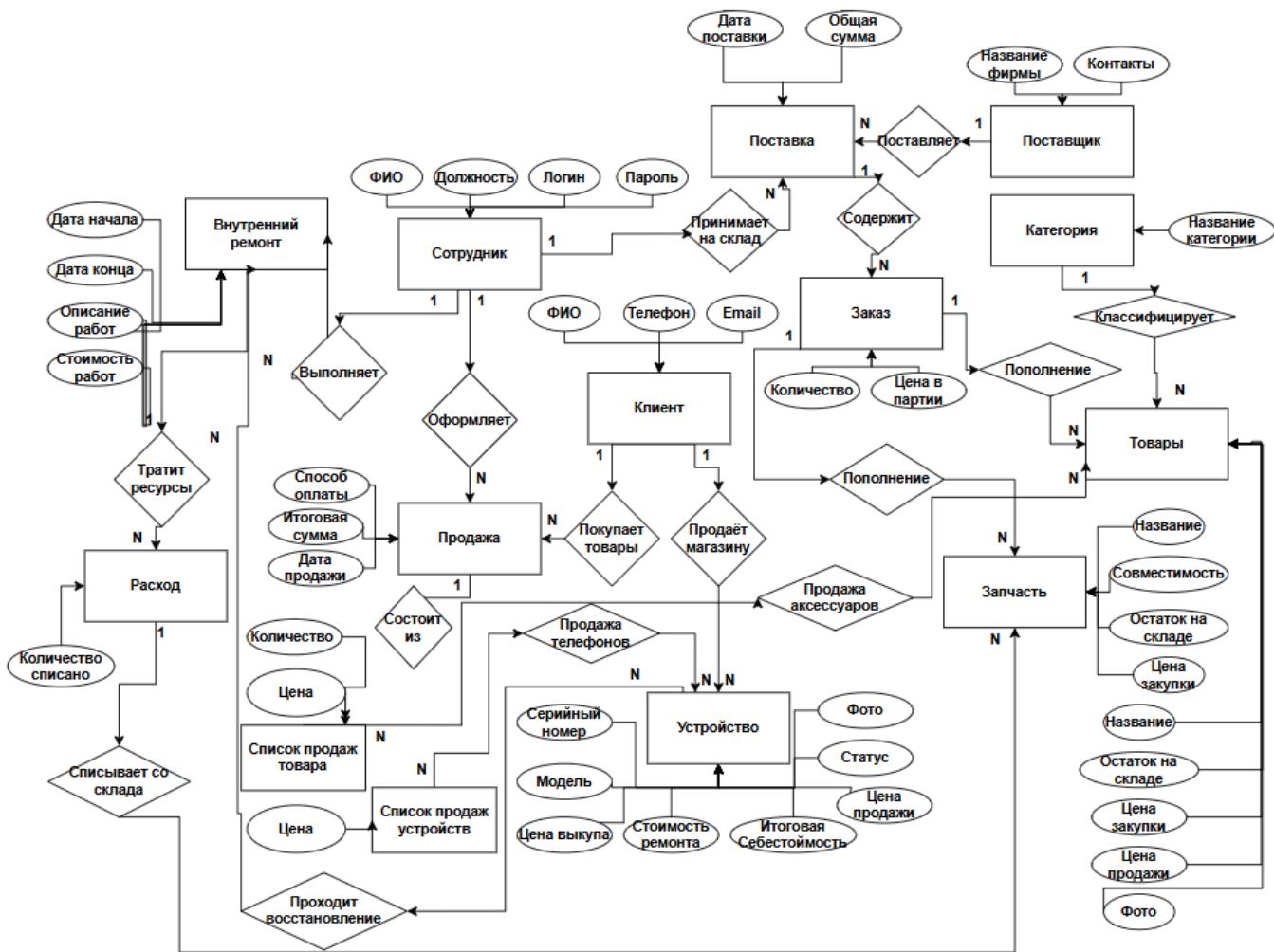


Рисунок 1.2 – Концептуальная модель данных

Изм.	Лист	№ докум.	Подп.	Дата

Приложение 2 Текст программы

<https://github.com/KSUSISS/KursivayaSUBD/tree/main>

```
DECLARE
    part_cost DECIMAL(10,2);

BEGIN
    -- 1. Узнаем цену запчасти
    SELECT purchase_price INTO part_cost FROM parts WHERE part_id = NEW.part_id;

    -- 2. Списываем со склада
    UPDATE parts SET stock_quantity = stock_quantity - NEW.quantity WHERE part_id = NEW.part_id;

    -- 3. Добавляем стоимость к телефону (через таблицу Ремонта -> Устройства)
    UPDATE devices
    SET repair_costs = repair_costs + (part_cost * NEW.quantity)
    WHERE device_id = (SELECT device_id FROM internal_repairs WHERE repair_id = NEW.repair_id);

    RETURN NEW;
END;

BEGIN
    IF NEW.accessory_id IS NOT NULL THEN
        UPDATE accessories
        SET stock_quantity = stock_quantity - NEW.quantity
        WHERE accessory_id = NEW.accessory_id;
    ELSIF NEW.device_id IS NOT NULL THEN
        UPDATE devices
        SET status = 'sold'
        WHERE device_id = NEW.device_id;
    END IF;
    RETURN NEW;
END;
```

Изм.	Лист	№ докум.	Подп.	Дата

```

BEGIN
    IF NEW.part_id IS NOT NULL THEN
        UPDATE parts
        SET stock_quantity = stock_quantity + NEW.quantity,
            purchase_price = NEW.purchase_price -- Обновляем цену закупки на
последнюю
        WHERE part_id = NEW.part_id;
    ELSIF NEW.accessory_id IS NOT NULL THEN
        UPDATE accessories
        SET stock_quantity = stock_quantity + NEW.quantity,
            purchase_price = NEW.purchase_price
        WHERE accessory_id = NEW.accessory_id;
    END IF;
    RETURN NEW;
END;

```

Изм.	Лист	№ докум.	Подп.	Дата

Приложение 3 Скриншоты программы

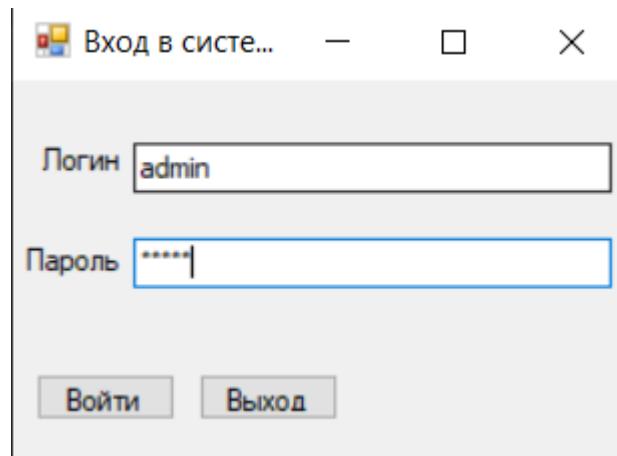


Рисунок 3.1 – Вход в программу

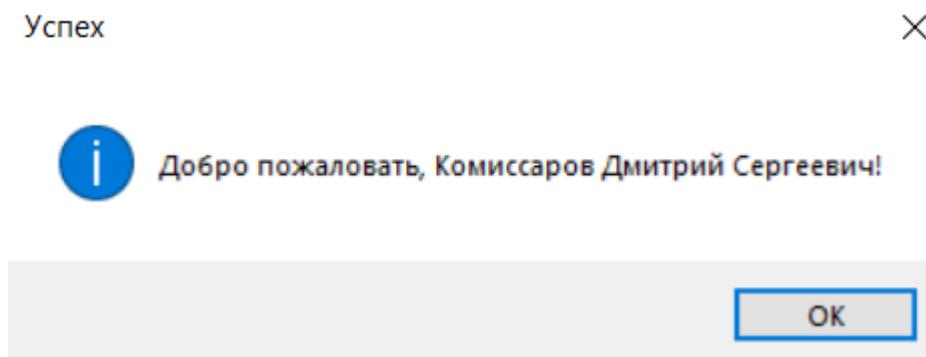


Рисунок 3.2 – Успешный вход

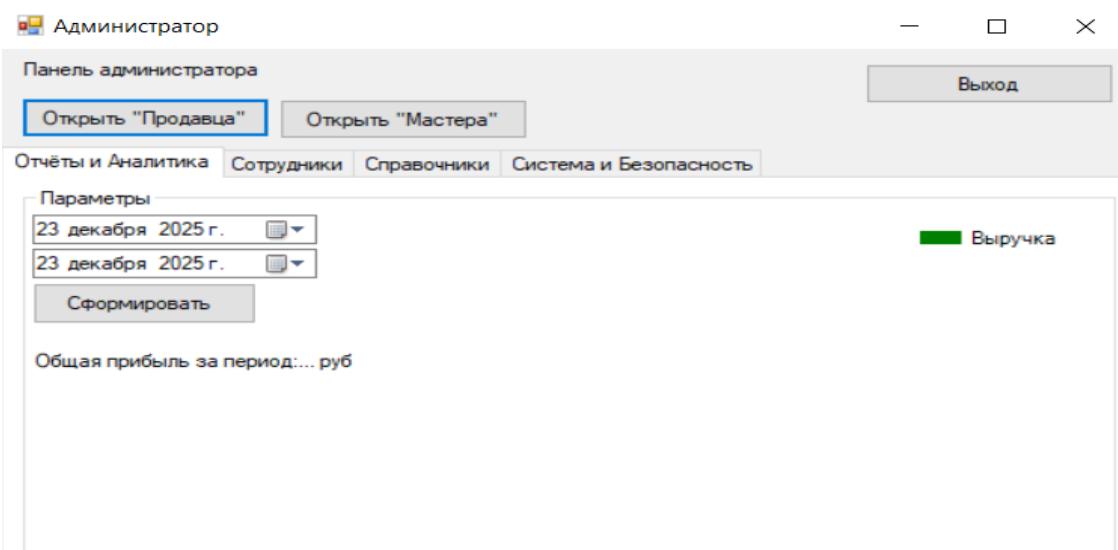


Рисунок 3.3 – Администратор. Владка Отчёты и Аналитика

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	
					МИВУ 09.03.04 ПЗ
					35

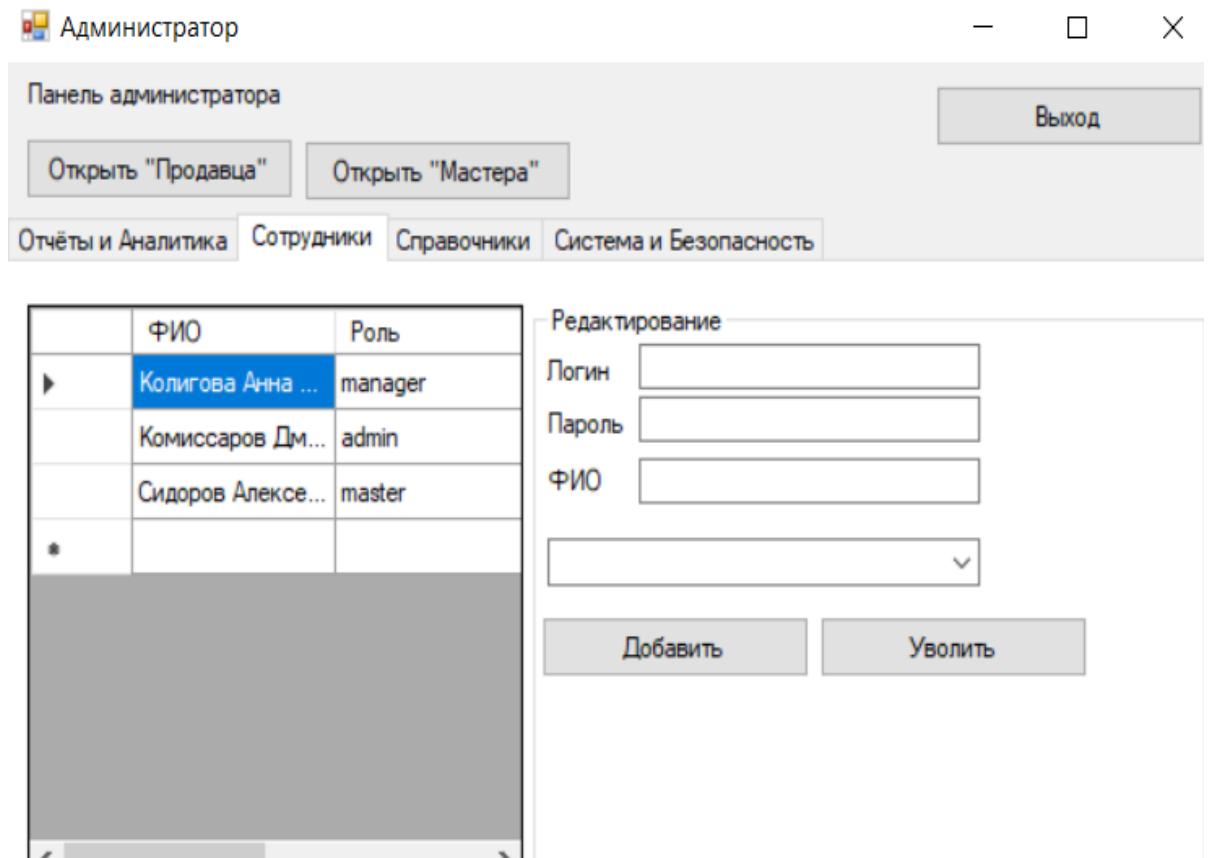


Рисунок 3.4 – Вкладка Сотрудники

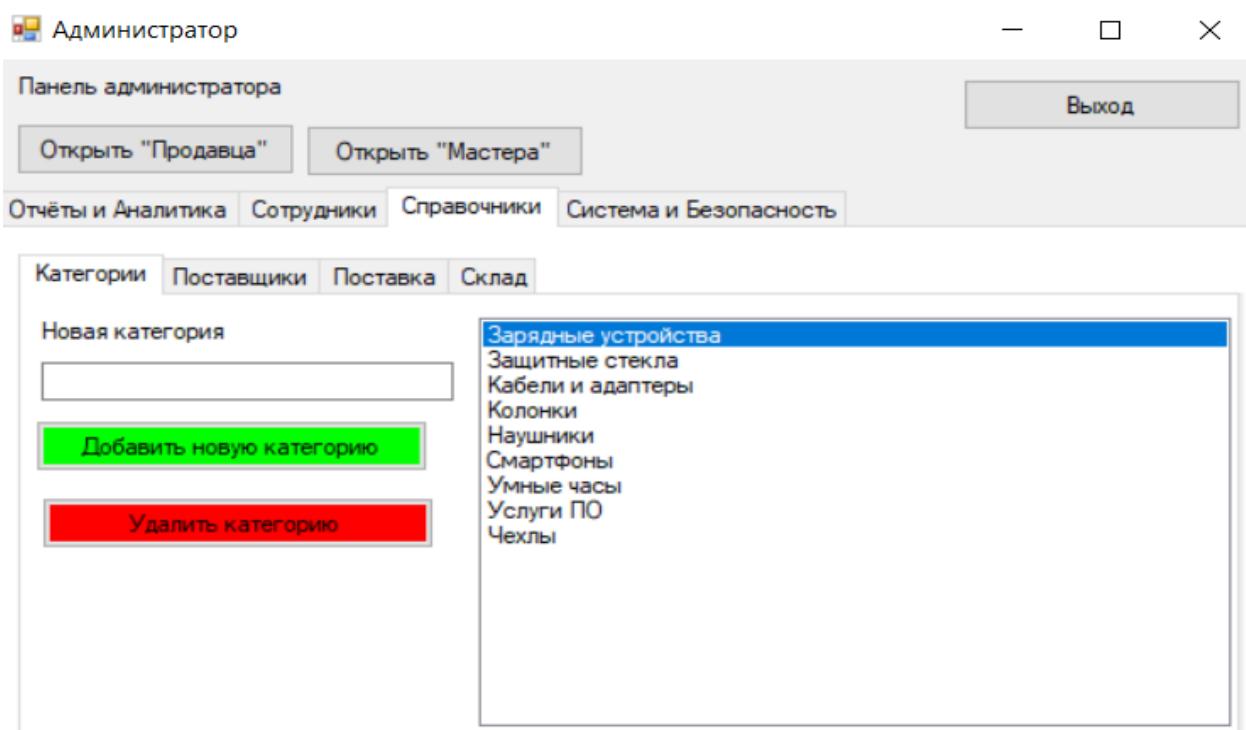


Рисунок 3.5 – Вкладка Справочники (Категории товаров)

Изм.	Лист	№ докум.	Подп.	Дата

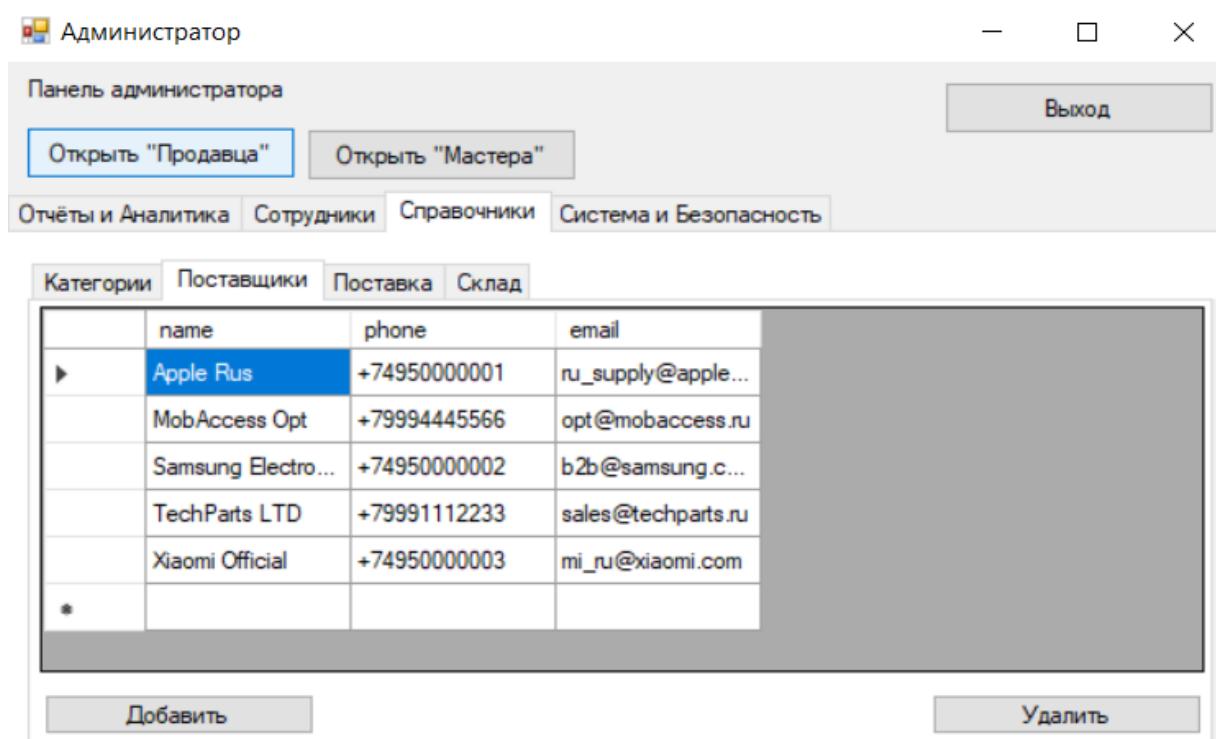


Рисунок 3.6 - Вкладка Справочники (Поставщики)

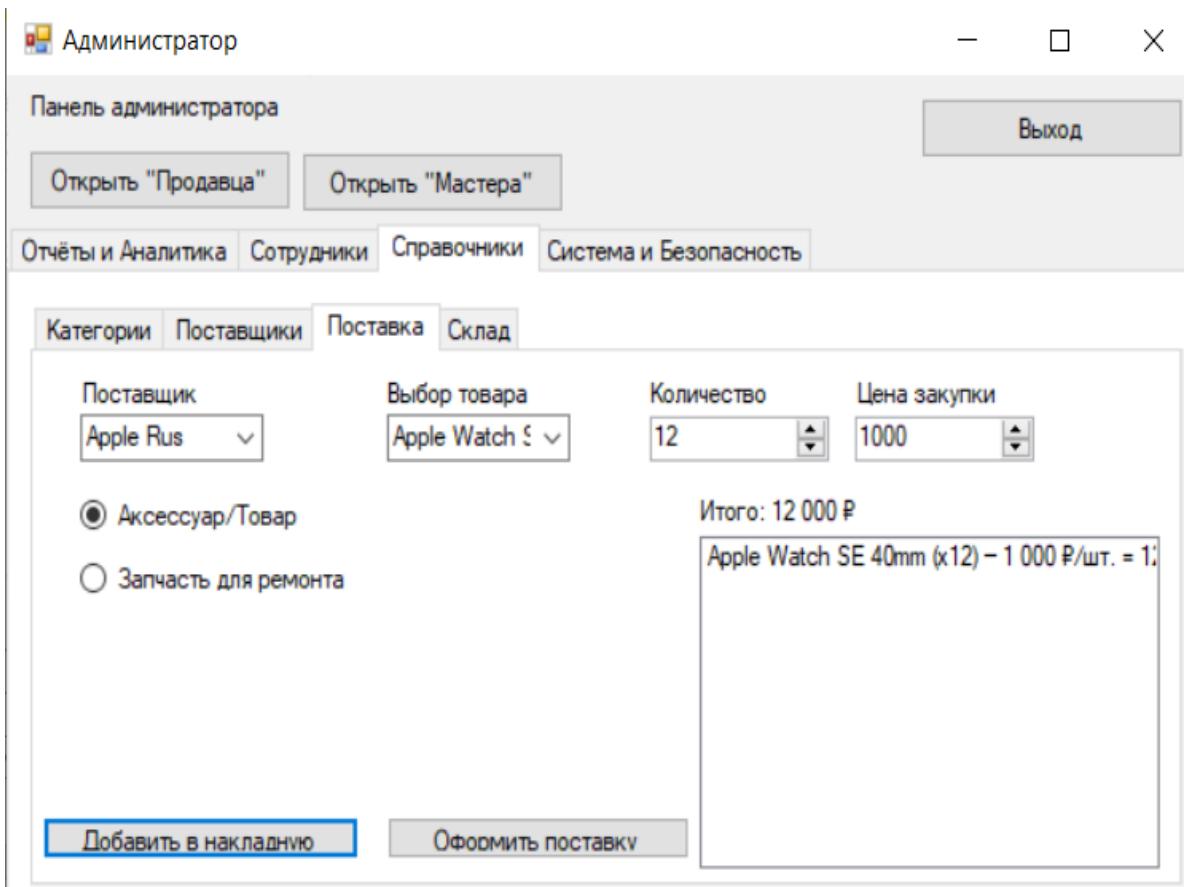


Рисунок 3.7 - Вкладка Справочники (Поставка)

Изм.	Лист	№ докум.	Подп.	Дата	Лист
					МИВУ 09.03.04 ПЗ

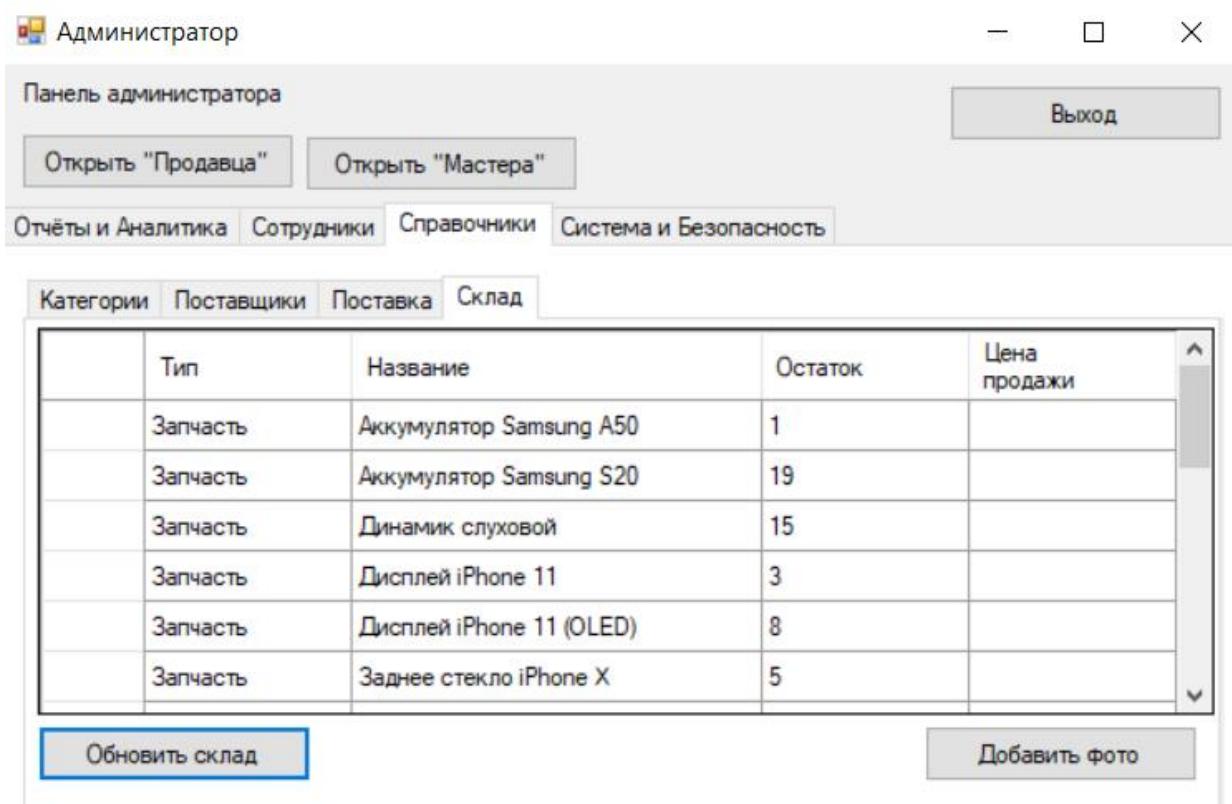


Рисунок 3.8 - Вкладка Справочники (Склад)

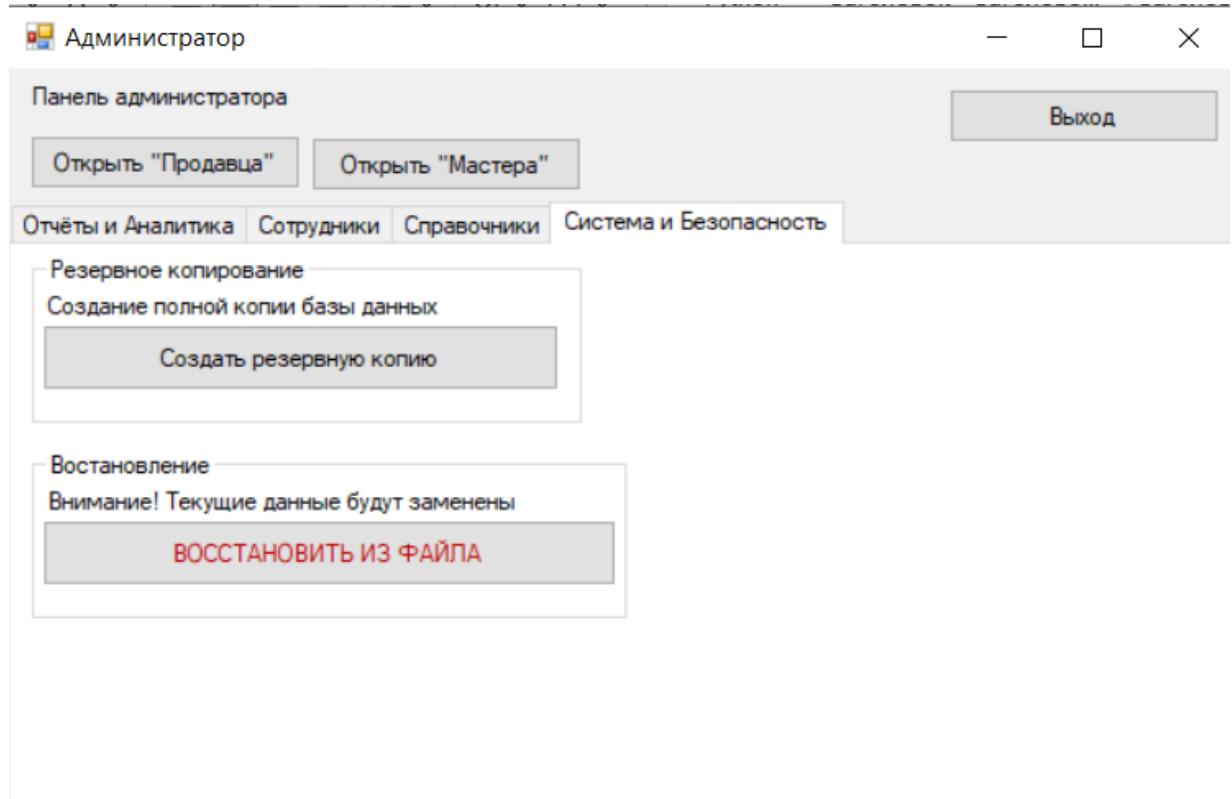


Рисунок 3.9 – Вкладка Система и безопасность

Изм.	Лист	№ докум.	Подп.	Дата	Лист	38
					МИВУ 09.03.04 ПЗ	

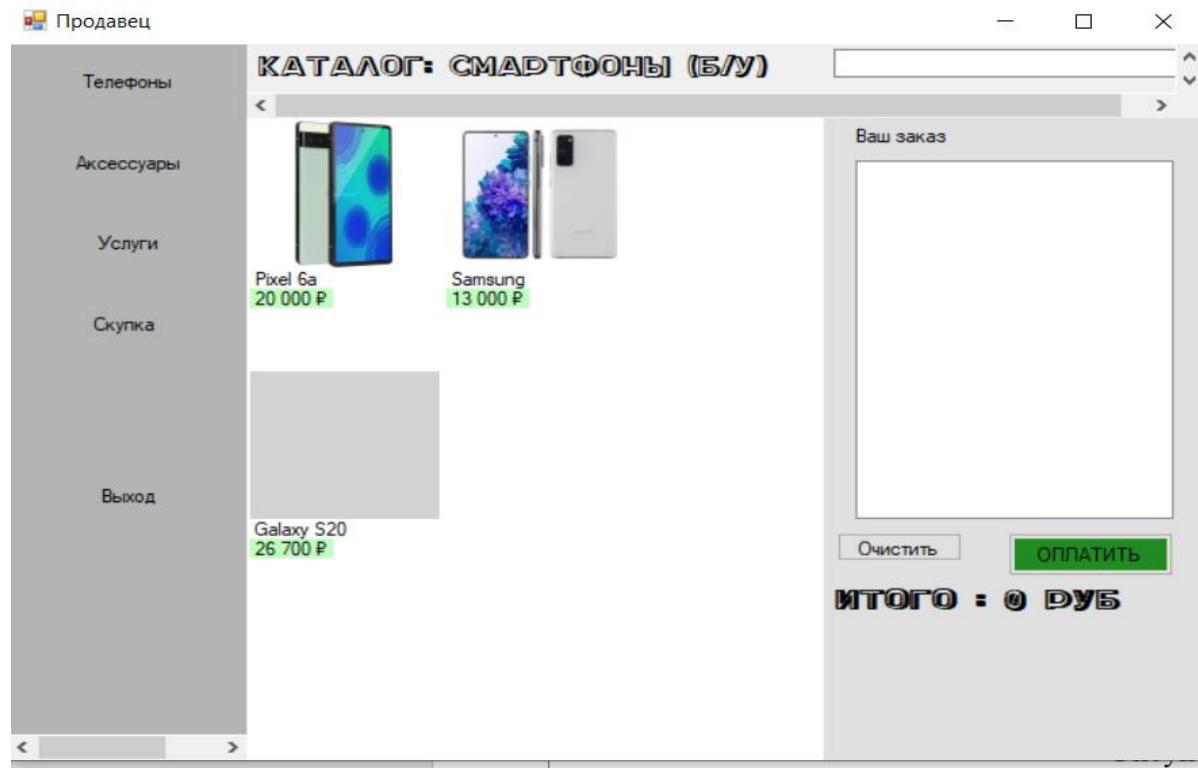


Рисунок 3.10 – Продавец

Рисунок 3.11 – Скупка

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	
					МИВУ 09.03.04 ПЗ
					39

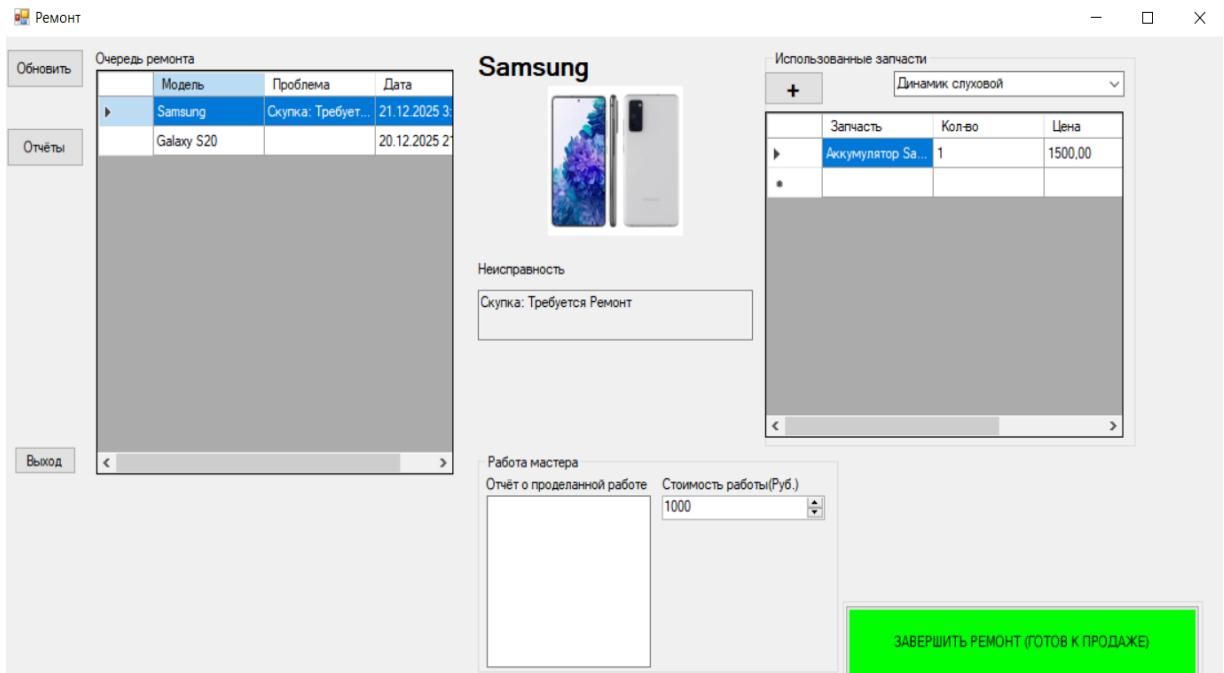


Рисунок 3.12 – Мастер

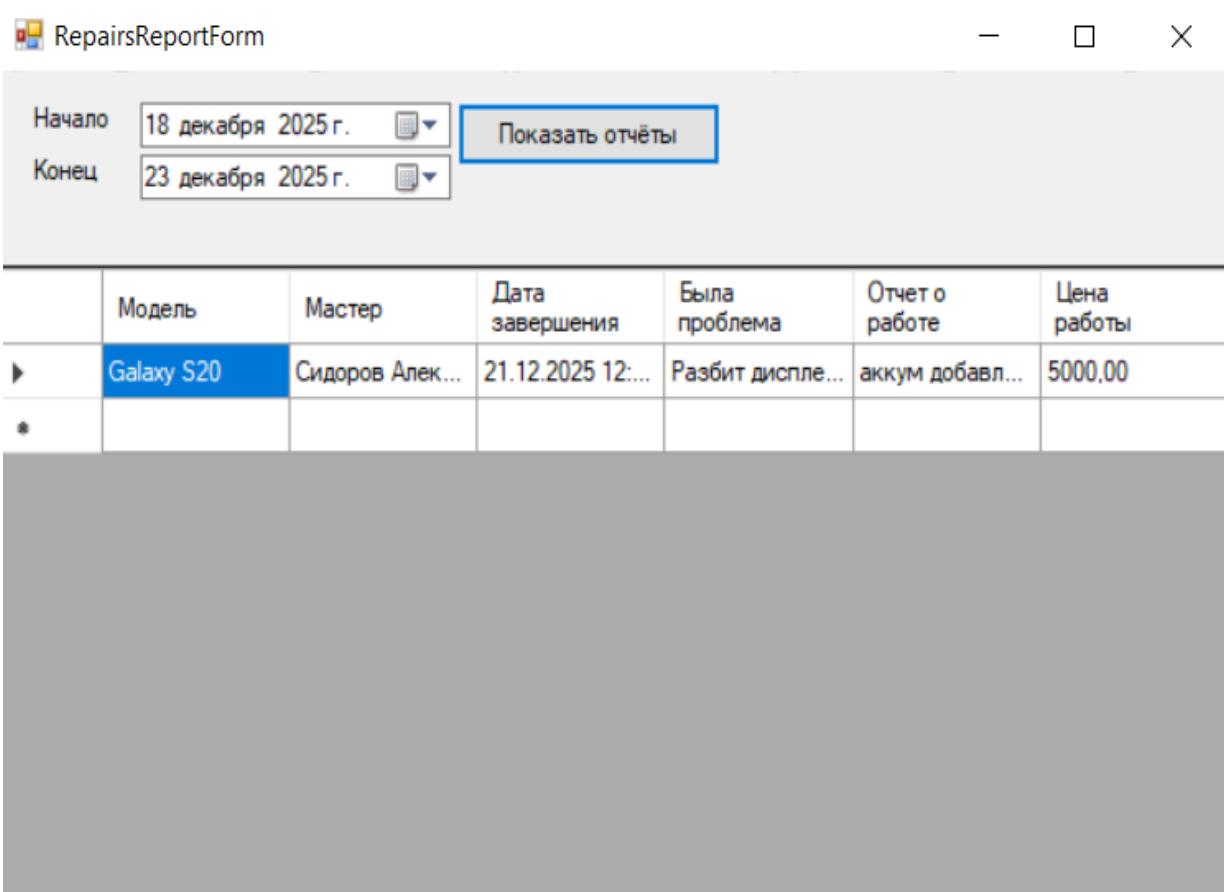


Рисунок 3.13 – Отчёты мастера

					Лист
Изм.	Лист	№ докум.	Подп.	Дата	
					МИВУ 09.03.04 ПЗ
					40