

**КОЛЛЕДЖ КОСМИЧЕСКОГО МАШИНОСТРОЕНИЯ И ТЕХНОЛОГИЙ**

|  |
| --- |
|  |

«Допущен к защите»

Председатель цикловой комиссии \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Эшанов

«\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г.

**РАЗРАБОТКА АРМ «РАБОЧЕЕ МЕСТО КАССИРА»   
ДЛЯ ФИРМЫ «НЕОНКА»**

**ДП.09.02.03.19.08ПЗ**

**Дипломный проект**

**Королёв, 2020**



**КОЛЛЕДЖ КОСМИЧЕСКОГО МАШИНОСТРОЕНИЯ И ТЕХНОЛОГИЙ**

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | «УТВЕРЖДАЮ»  Заместитель директора колледжа  по учебной работе  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.В.Антропова  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. | |

**ЗАДАНИЕ**

**на дипломный проект**

студенту группы П2-16 **Нишанову Виктору Троадиевичу**

Тема задания «**Разработка АРМ «Рабочее место кассира»**

**Для фирмы «Неонка»»** утверждена приказом по университету от «\_\_\_» апреля 2020 г. № \_\_\_\_\_\_

Руководитель дипломного проекта Г**усятинер Л.Б.**

**Содержание работы:**

Введение.

Глава 1. Теоретическая часть

Глава 2. Проектная часть

Глава 3. Организационная-экономическая часть

Заключение.

Список использованной литературы.

Приложения.

Презентация.

Срок выполнения дипломного проекта 15 июня 2020 года

Председатель цикловой комиссии 09.02.03 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, Ф.И.О.

Руководитель дипломной работы Гусятинер Л. Б.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись студента

Дата выдачи задания «\_\_\_\_» апреля 2020 года

**СОДЕРЖАНИЕ**

*Изм.*

*Лист*

*№ докум.*

*Подпись*

*Дата*

*Лист*

*3*

**ДП.09.02.03.19.08П3**

*Разработал*

Нишанов В.Т.

*Консультант*

Гусятинер Л. Б.

*Рецензент*

Конс.поэк.ч

Норм-контр.

РАЗРАБОТКА АРМ «РАБОЧЕЕ МЕСТО КАССИРА»

ДЛЯ ФИРМЫ «НЕОНКА»

*Лит.*

*Листов*

19

Группа П2-16

[ВВЕДЕНИЕ 2](#_Toc41764250)

[1. Теоретическая часть 3](#_Toc41764251)

[1.1. О рабочем месте кассира 3](#_Toc41764252)

[1.2. Математическая постановка задачи 4](#_Toc41764253)

[1.3. Структура рабочего места кассира 5](#_Toc41764254)

[1.4. Анализ организации учета оборудования и компьютерной техники 8](#_Toc41764255)

[1.4. Анализ имеющихся программных решений 10](#_Toc41764256)

[1.4.1. Frontol [1] 10](#_Toc41764257)

[1.4.2. АРМ кассира Стандарт [2] 11](#_Toc41764258)

[1.4.3. Штрих-М: Кассир [3] 13](#_Toc41764259)

[1.4.4. 1С Розница [4] 14](#_Toc41764260)

[1.4.5. Тирика-магазин [5] 17](#_Toc41764261)

[2. Проектная часть 19](#_Toc41764262)

[2.1. Алгоритмы решения 19](#_Toc41764263)

[2.2. Выбор инструментов, сред, языков 22](#_Toc41764264)

[2.3. Структура программного комплекса 23](#_Toc41764265)

[2.3.1. Логическое и физическое моделирование базы данных 23](#_Toc41764266)

[2.3.2. Разработка автоматизированной системы 24](#_Toc41764267)

[2.3.3. Описание тестовых наборов 28](#_Toc41764268)

[2.4. Описание процедур и функций 30](#_Toc41764269)

# ВВЕДЕНИЕ

Задачей дипломного проекта является разработка программы для рабочего места кассира. Автоматизированная система рабочего места кассира по продаже мебели должна выполнять следующие функции: работу с оплатой/возвратом какой-нибудь мебели, печать квитанции, а также систему авторизации и регистрации кассира.

В теоретической части идет описание о рабочем месте кассира, какие задачи он выполняет, какое оборудование для него требуется. Также в теоретической части рассказано о математической постановке задачи – в ней приведены формулы по расчету мебели.

Во второй главе опишем структуру базы данных проекта, а также построим структуру интерфейса пользователей, разработаем алгоритмы основных функций системы. Следующим шагом станет определение требований к автоматизации на основе выделенных бизнес-процессов. Данные требования должны будут предъявляться к функциональным возможностям системы, к документообороту и аналитическим расчетам, к интерфейсу системы.

В третьей главе опишем программное обеспечение, необходимое для эксплуатации программы, рассчитаем показатели экономической эффективности.

Так как дипломный проект посвящен разработке автоматизации рабочего места кассира, то целью будет являться разработка веб-интерфейса для автоматизации рабочего места кассира. Для разработки потребуется выполнить следующие задачи:

* Провести анализ существующих систем, чтобы максимально грамотно внедрить программный продукт.
* Изучить процессы, документы, и информацию подлежащие автоматизации.
* Спроектировать программную модель для приложения.
* Спроектировать графический интерфейс.
* Написание кода.

# Теоретическая часть

## О рабочем месте кассира

Рабочее место кассира – это пространство в торговом зале, где установлено оборудование для автоматизации продаж. Оснащение кассовой зоны такой техникой позволяет не только оптимизировать товаропоток, увеличив тем самым прибыль торговой точки, но и обезопасить процесс.



Рис. 1. Рабочее место кассира

Оборудование рабочего места кассира в магазине может включать:

* POS-систему – компактный компьютерный терминал;
* кассовый аппарат (или фискальный регистратор);
* сканер штрихкода;
* дисплей покупателя;
* денежный ящик;
* программируемая клавиатура;
* считыватель магнитных карт;
* весовое оборудование;
* монитор;
* детектор банкнот;
* кассовое программное обеспечение
* платежный терминал для приема банковских карт.

Выбор оснащения зависит от специфики товара, выделенного пространства и объемов продаж.

Автоматизировать процесс в небольшой торговой точке с развесной продукцией можно с помощью комплекта «кассовый аппарат + сканер + весы». Для алкогольной товарной группы понадобится 2D-сканер и кассовое оборудование с ПО, совместимым с ЕГАИС.

Организация рабочего места кассира требует стартовых вложений, однако именно от технической оснащенности зависит интенсивность и безопасность товарооборота, а также репутация магазина среди покупателей.

## **Математическая постановка задачи**

Задача: на квитанции в фирме «Неонка» указаны следующие категории: Цена за мебель, кол-во. Требуется посчитать суммы за 1 и более товаров.

1. Формула вычисления итоговой суммы за 1 товар:

***Цена за мебель \* кол-во = Итого***

1. Формула вычисления итоговой суммы 2 товаров:

***Цена за мебель \* кол-во + Цена за мебель \* кол-во = Итого***

1. Формула вычисления итоговой суммы за несколько товаров:

***Цена за мебель \* кол-во + Цена за мебель \* кол-во + … + Цена за мебель \* кол-во = Итого***

Пример:

Допустим, покупатель заказал 4 стула, 1 диван и 2 кровати. 1 стул, например, стоит 15000 рублей, а 1 диван - 39000 рублей, а 1 кровать – 25000. Посчитаем сумму по формулам, а затем впишем их в 1 таблицу.

1. 15000 \* 4 = 60000 (руб.)
2. 39000 \* 1 = 39000 (руб.)
3. 25000 \* 2 = 50000 (руб.)
4. 60000 + 39000 + 50000 = 149000 (руб.)

Таблица 1. Пример подсчета Квитанции

|  |  |  |  |
| --- | --- | --- | --- |
| Мебель | Цена | Кол-во | Сумма |
| Стул | 15000 рублей | 4 шт. | 60000 рублей |
| Диван | 39000 рублей | 1 шт. | 39000 рублей |
| Кровать | 25000 рублей | 2 шт. | 50000 рублей |
| Итого: | | | 149000 рублей |

### Структура рабочего места кассира

Современное место кассира уже давно обладает системами автоматизации для увеличения производительности. Наличие современной компьютерно-кассовой системы позволяет вывести торговлю на принципиально новый уровень путём создания автоматизированного рабочего места кассира. Устаревшие кассовые аппараты уже не выдерживают конкуренции с новым оборудованием: они менее функциональны, а значит – неудобны.

Расширить функционал большинства старых моделей практически невозможно, в то же время конкуренты уже приобретают прогрессивные системы с новейшим программным обеспечением. Экономия в данном случае становится просто неуместной, поскольку рынок диктует свои условия. Новейшие компьютерно-кассовые комплексы позволяют быстро и качественно решать множество задач: от непосредственного обслуживания клиентов до формирования кассовых отчётов.

Основными процессами рабочего места кассира являются:

* Авторизация, регистрация кассира;
* Добавление товара в корзину;
* Подсчет общей суммы за товары;
* Оплата товара;
* Возврат товара;

Проведение оплаты/возврата товара осуществляется при помощи рабочего места кассира. Допустим покупатель выбрал какую-нибудь мебель и должен ее оплатить. Кассир добавляет эту информацию в свою программу и делает запрос на оплату, после чего покупатель оплачивает наличными или картой за мебель.

Основной целью продавца-кассира является проведение расчетов с покупателями в магазине. Для качественного достижения поставленной цели продавец-кассир обязан выполнять следующие основные функции:

* подготовка к работе. Кассир должен подготовить свое рабочее место, авторизоваться в программе на компьютере, прежде чем начнет работу;
* обслуживание покупателей. Кассир обязан давать консультацию по мебели покупателю, оплачивать товар на кассе и выдавать гарантийный талон с квитанцией об оплате;
* сдача выручки. Кассир обязан отчитаться о проделанной работе своему руководителю или начальнику;
* завершение работы. Кассир при завершении работы должен перед уходом выйти из своего аккаунта на компьютере и прибраться на своем рабочем месте.

Процесс покупки мебели состоит из четырех основных операций:

* размещение мебели в корзину;
* подсчет суммы и итого за мебель;
* оплата за мебель наличными или картой;
* получение квитанции и гарантийного талона;
* текущий учет в процессе хранения.

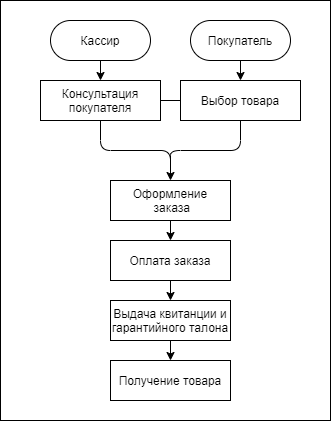


Рис. 2. Структура оплачивания товаров

## Анализ организации учета оборудования и компьютерной техники

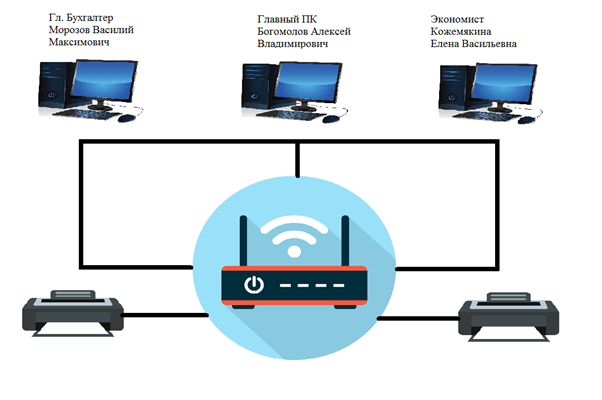


Рис. 2.1. Блок-схема аппаратного обеспечения

Аппаратное обеспечение «Фирма Нионка» состоит из:

1. 3 Пользователя ЛВС

2. 2 Принтера

3. 1 Коммутатор

Пользователи ЛВС используют стационарные ПК с системными блоками, имеющими следующие характеристиками:

Таблица 1. Характеристики ПК пользователей ЛВС

|  |  |  |  |
| --- | --- | --- | --- |
| Должность пользователя | Наименование системного блока ПК | Характеристики | |
| Главный бухгалтер | Компьютер HP Desktop Pro | Процессор | Intel-core i3-6100 |
| Оперативная память | 4GB RAM |
| Видеокарта | Nvidia GeForce GTX 650 |
| Жесткий диск | 120GB SSD |
| Блок питания | 3Cott-450ATX с 450 WT |
| Экономист | Компьютер HP Desktop Pro | Процессор | Intel-core i3-9100f |
| Оперативная память | 8GB RAM |
| Видеокарта | Nvidia Geforce GTX 1050 |
| Жесткий диск | 480GB SSD |
| Блок питания | 600WT |
| Программист | Компьютер HP Desktop Pro | Процессор | Intel-core i5-4460 |
| Оперативная память | 8GB RAM |
| Видеокарта | Nvidia GeForce GTX 1060 |
| Жесткий диск | 480GB SSD |
| Блок питания | 600WT |

Также пользователи имеют:

* 3 монитора 15.6 ASUS VT168N;
* 3 мышки Logitech G102 Prodigy;
* 3 клавиатуры HyperX Alloy FPS PRO HX-KB4RD1-RU/R1;
* Маршрутизатор Keenetic Ultra KN-1810;
* 2 Принтера - лазерный Brother HL-1202R.

## 1.4. Анализ имеющихся программных решений

Прежде, чем приступить к проектированию необходимо обосновать целесообразность предстоящей разработки. В настоящий момент на рынке можно найти большое количество аналогичных по профилю программных продуктов.

### 1.4.1. Frontol [1]

Frontol — одна из современных кассовых программ, производителем которой является фирма «АТОЛ-Технологии» класса Front-Office для автоматизации торгового зала предприятия розничной торговли и общественного питания различных форматов и размеров, применяется в кассовых системах на базе фискального регистратора. Данный программный продукт совместим с системой программ 1С ЕГАИС.



Рис. 3.1. Логотип Frontol

Frontol поддерживает работу с широким «ассортиментом» торгового оборудования. Продукт создан как клиент-серверный, использует СУБД FireBird 2.0, с существенной увеличенной скоростью работы ПО при немалом объеме данных в сочетании с высокой надежностью. Есть возможность удаленного администрирования кассовых рабочих мест. С использованием встроенного генератора отчетов FastReport могут быть самостоятельно разработаны любые печатные формы документов: товарный чек, перемещение, инвентаризация, счет-фактура и т.д.

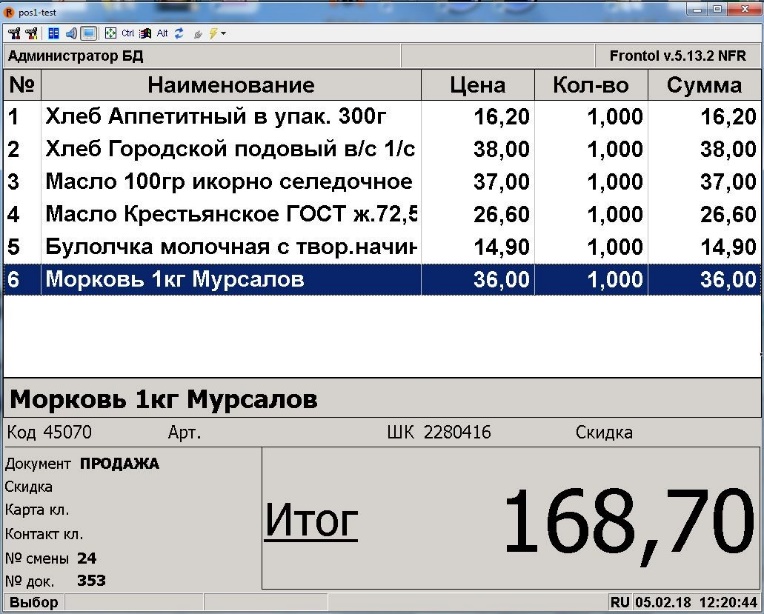


Рис. 3.2. Пример работы программы «Frontol»

В зависимости от области применения, Frontol поставляется в следующих вариантах: Лайт, Стандарт, Оптим, Супермаркет, Бутик, Торговля, Фастфуд и Кафе. Установка ЕГАИС потребует соответствующей версии кассовой программы.

### 1.4.2. АРМ кассира Стандарт [2]



Рис. 3.3. Логотип ККС

АРМ Кассира – одна из современных программ для работы с кассой «ККС» предлагает кассовую программу АРМ Кассира в трех версиях: лайт, стандарт и проф. Эти версии отличаются функционалом: версия Лайт – предоставляет минимальный функционал, обеспечивающий эффективную работу маленького магазина или бутика. Эта версия не содержит ничего лишнего, так что начать работу с ней будет просто.

АРМ Кассира Стандарт - кассовая программа с расширенным функционалом для супермаркета самообслуживания с торговой площадью до 1500 м2. Версия Проф предоставляет максимальные возможности для эффективной организации работы крупных торговых предприятий, супермаркетов, в том числе сетевых магазинов. Версии кассовой программы от ККС отличаются по стоимости в зависимости от функциональности. Если вы открываете свой первый небольшой магазин, то рекомендуем вам начать с недорогой версии Лайт и затем, по мере расширения, перейти к более дорогим Стандарт и Проф.

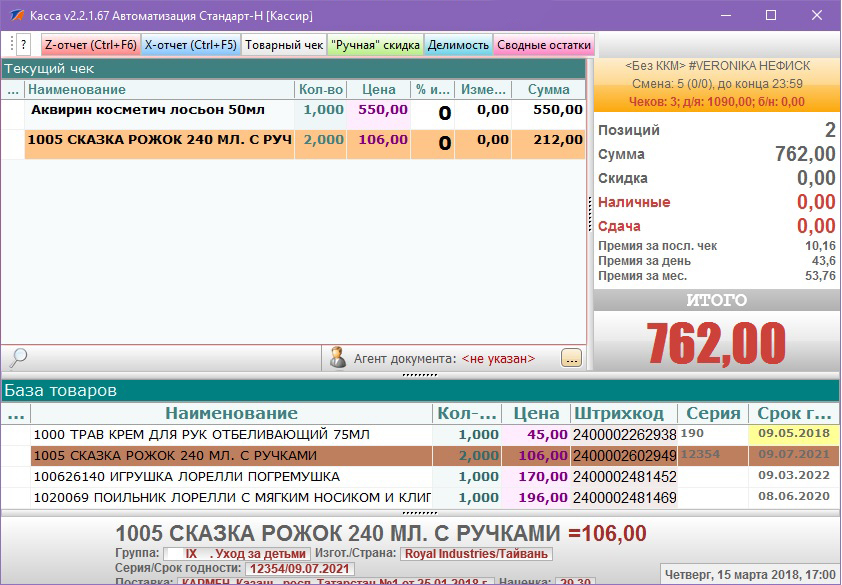


Рис. 3.4. Пример работы программы «АРМ Кассира Стандарт»

ККС рассмотрим преимущества кассовых программ от ККС. АРМ Кассира может быть использована не только в магазинах розничной торговли и супермаркетах, но и на предприятиях быстрого питания, организациях с прилавочной торговлей, в кафе, барах, ресторанах. Кассовая программа АРМ кассира позволяет подключать для работы широкий диапазон дополнительного POS-оборудования, например, фискальные регистраторы, дисплеи покупателей, программируемые клавиатуры, сканеры штрих-кодов, считыватели магнитных карт.

Одним из основных преимуществ программы АРМ Кассира является совместимость с системой 1С: Предприятие. В качестве хранилища информации выступает клиент-серверная СУБД, которая обеспечивает целостность, сохранность и непротиворечивость данных, а также защиту при нештатных ситуациях (например, при отключении электричества). Система обладает модульной архитектурой, которая позволяет подключать различные компоненты (в том числе от сторонних производителей), есть возможность подключения нескольких фискальных регистраторов.

Кассовые программы от ККС позволяют усовершенствовать работу франчайзинга и мульти брендовых магазинов, печатать чеки от различных юридических лиц на разных фискальных регистраторах. Система поддерживает разделение прав пользователей по группам. Администратор может назначать наборы прав пользователям непосредственно с рабочего места, не отходя от кассы. Благодаря функции "отложенный документ" можно избежать очередей - считывание кодов товаров происходит с помощью терминалов сбора данных, а на кассе производится окончательный расчет.

### 1.4.3. Штрих-М: Кассир [3]

«Штрих-М: Кассир» v.5 — пакет прикладных программ, предназначенный для решения задачи автоматизации рабочего места кассира на предприятиях торговли. Конфигурация «Штрих-М: Кассир» v.5 позволяет решать задачу автоматизации рабочего места кассира на любых предприятиях розничной торговли с использованием широкого спектра торгового оборудования.



Рис.3.5. Логотип Штрих-М

Штрих-М конфигурация «Штрих-М: Кассир v .5.0» представляет собой самостоятельную конфигурацию платформы «1C: Предприятие v . 8.0». Программа обладает простым и дружественным интерфейсом для кассира и администратора. Система «Штрих-М: Кассир v.5.0» может обмениваться данными с широким спектром товароучетных программ, например с типовой конфигурацией «1C: Торговля и склад v.7.7», «Штрих-М: Торговое предприятие v.5.0» и многими другими. Благодаря открытому протоколу обмена, подключение к любой товароучетной системе требует минимальных доработок.

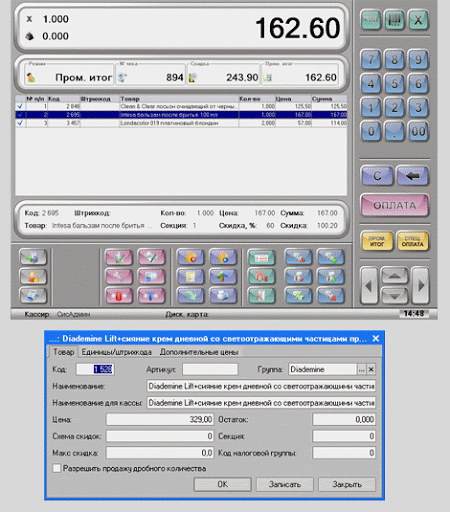


Рис.3.6. Пример работы программы «Штрих-М: Кассир v .5.0»

Кассовая программа «Кассир» поддерживает оплату покупок с помощью банковских карт. Программа позволяет расширять функционал конфигурации даже после запуска предприятия, обладает простым и дружественным интерфейсом как для кассира, так и для администратора. Конфигурация «Штрих-М: Кассир v.5.0» поддерживает работу с сенсорным экраном благодаря технологии TouchScreen, за счет чего максимально упрощается обучение и работа персонала.

### 1.4.4. 1С Розница [4]

1С Розница – это программа для онлайн-кассы магазина, поддерживающая требования 54-фз:

* наименование документа;
* порядковый номер за смену;
* дата, время и место (адрес) осуществления расчета (при расчете в зданиях и помещениях - адрес здания и помещения с почтовым индексом, при расчете в транспортных средствах - наименование и номер транспортного средства, адрес организации либо адрес регистрации индивидуального предпринимателя, при расчете в сети "Интернет" - адрес сайта пользователя);
* наименование организации-пользователя или фамилия, имя, отчество (при наличии) индивидуального предпринимателя - пользователя;
* идентификационный номер налогоплательщика пользователя;
* применяемая при расчете система налогообложения;

В программе "1С Розница" есть режим рабочего места кассира для работе на кассе. В этом режиме программа может использоваться для учета продаж в магазине с использованием сканера штрихкодов или без него.



Рис 3.7. 1С розница

В режиме работы автоматизированного рабочего места кассира на экране отображается только информация для учета продаж.

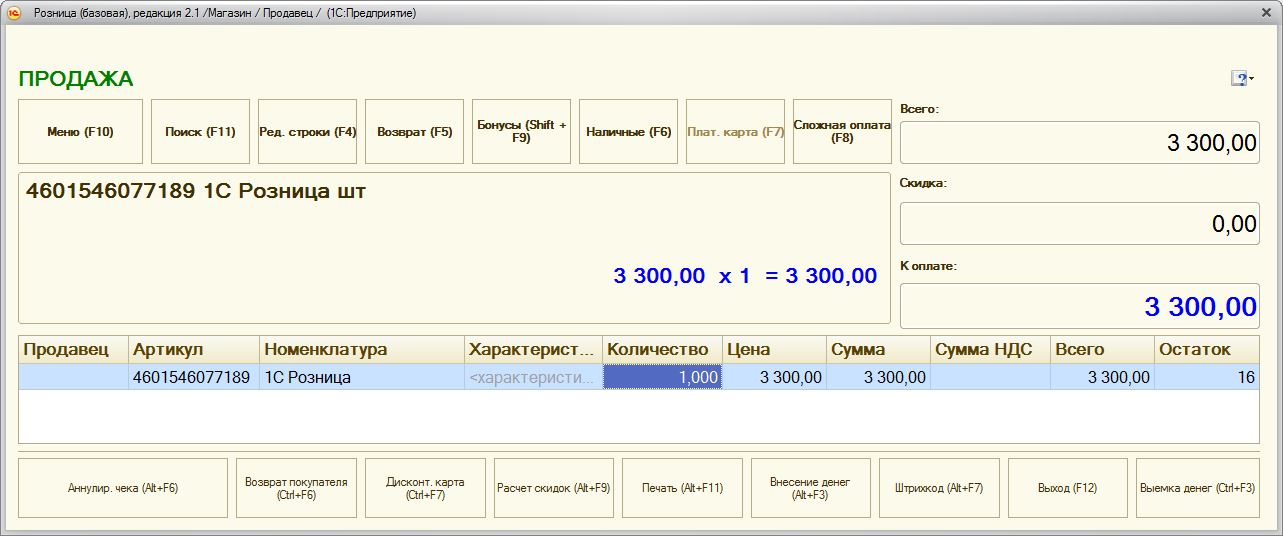


Рис. 3.8. Информация для учета продаж

Выбор товаров и услуг в чек в программе "1С Розница" может осуществляться несколькими способами:

* С использованием [сканера штрих кода](https://sevsoft.ru/services/podkluchenie-torgovogo-oborudovaniya-1c.shtml#scanner)

Самый удобный и быстрый способ заполнения чека. Идеальный вариант для товаров с нанесенным штрихкодом.

* Выбор товара, услуги из перечня товаров(справочника)

На экране отображается полный перечень товаров и услуг. Этот способ используется если, например штрих-код не читается или отсутствует, а точное название товара кассир не помнит.

* Выбор товара, услуги по коду

На клавиатуре вводится 5 цифр кода товара или услуги. Удобно при отсутствии сканера штрихкодов и небольшом перечне товаров или услуг.

* Выбор товара, услуги по названию

Программа ищет товары в справочнике по названию. Вам не нужно набирать полностью название товара. Такой вариант часто используется для учета продажи УСЛУГ

* Расширенный поиск товара по нескольким полям, включая остаток и цену

При большом количестве товаров поиск можно осуществить сразу по нескольким полям. Например, только товары в наличии и с ценой 100 руб

* С использованием [терминала сбора данных](https://sevsoft.ru/catalog/tsd/)

Такой вариант используется если физически невозможно(или не нужно) поднести товар к кассе, но количество наименований товаров велико для набора вручную.

Кроме этого в программе "1С Розница" есть такие полезные функции:

* + Предоставление скидки (наценки) вручную и по дисконтной карте
  + Бонусные скидки
  + Подарочные сертификаты
  + Автоматический расчет сдачи покупателю
  + Печать чека на онлайн-кассе 54-ФЗ
  + [Печать товарного чека](https://sevsoft.ru/1c-predpriyatie/roznica/rmk/check.shtml)

При оплате чека возможны различные варианты: только наличными, платежной картой, банковским кредитом, подарочным сертификатом или сложный вид оплаты. Для часто используемых видов оплат предусмотрены отдельные кнопки, что позволяет ускорить оформление оплаты.

Помимо чека на проданные товары, кассир может распечатать дополнительные документы: товарный чек, гарантийный талон для товаров с посерийным учетом, последний слип-чек, копии пробитых чеков.

Доступная цена программы, простота освоения делают её незаменимым помощником индивидуального предпринимателя.

### 1.4.5. Тирика-магазин [5]

Программа Тирика-Магазин является одной из качественных средств автоматизации **рабочего места кассира**. При запуске программы продавец попадает в окно продажи программы, состоящее из двух колонок. В первой колонке размещена информация, касающаяся текущей продажи: наименование покупателя (важно для магазином, выдающих дисконтные карты клиентам; во всех остальных случаях указывать покупателя, разумеется, необязательно), списка уже выбранных покупателем товаров и кнопки управления продажей: кнопка печати товарного или кассового чека, кнопка Продать, кнопка отмены чека и т.п. Во второй колонке размещена номенклатура товаров магазина, разбитая для удобства на группы и подгруппы и снабженная поиском. Здесь же для случая работы со сканером штрих кодов размещается специальное окно, в котором кассир сканирует штрихкод товара для добавления его в продажу.

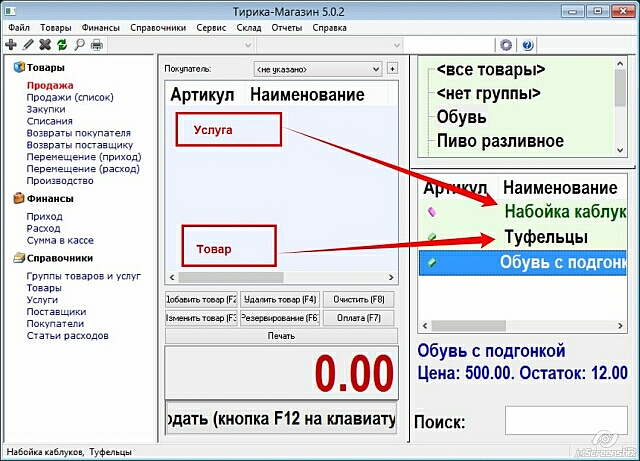


Рис. 3.10. Тирика-магазин

Когда на кассу приходит первый покупатель, кассир на своем рабочем месте добавляет выбранные покупателем продажи в чек. Если кассир использует при этом сканер штрихкодов, ему достаточно просто отсканировать штрихкод товара, после чего программа автоматически находит отсканированный товар в базе данных и открывает перед кассиром специальное окно, в котором он может указать количество товара. После нажатия кнопки OK в этом окне программа добавляет товар в нужном количестве в чек. Если же рабочее место кассира не оснащено сканером штрих кодов, кассир мышкой выбирает из списка нужный товар либо вводит его наименование либо артикул в окне поиска, и программа находит все товары с походими наименованиями и предлагает кассиру указать, какой именно товар из представленных он хочет добавить в продажу. кассир указывает нужный товар мышкой или клавиатурой, и программа точно так же открывает окно указания количества товара, после чего добавляет товар в указанном количестве в продажу.

В случае, если рабочее место кассира оснащено торговыми весами, программа автоматически считывает вес товара с весов, и выставляет его в окне указания количества товара, так что продавцу достаточно только подтвердить правильность взвешивания нажатием кнопки OK для того, чтобы добавить товар в продажу. После того, как весь выбранный покупателем товар будет добавлен в чек, кассир нажимает в программе кнопку Продать, программа запоминает выбранную продажу и автоматически печатает товарный либо кассовый чек в зависимости от того. какое именно оборудование для печати чеков подключено к рабочему месту кассира.

# Проектная часть

## Алгоритмы решения

На изображении 1 изображена Use Case диаграмма для программного продукта, который будет разработан в ходе выполнения дипломной работы:

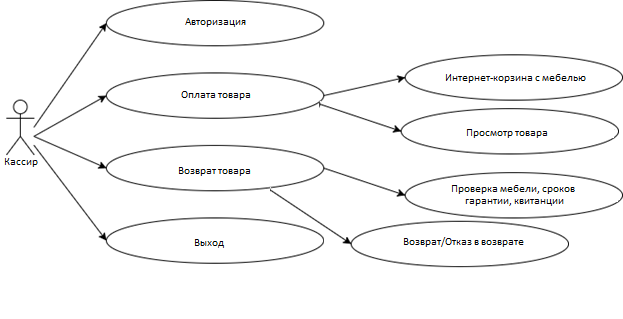


Рис 2.1.1 Use Case диаграмма рабочего места Кассира



Рис. 2.1.2 Use Case диаграмма рабочего места Администратора

Задача 1. Составление алгоритма работы программы.

Цель задачи: Алгоритм составляется на начальном этапе разработки проекта. Он предназначен для разбиения работы программы на более простые шаги, которые в последующем помогут решить поставленную задачу.

Функции: Объединение всех поставленных задач и целей в единую систему, с дальнейшим переносом на язык программирования.

Входные данные: кассир, рабочее место.

Выходные данные: Автоматизация рабочего места кассира

Задача 2. Выбор ПО для реализации проекта.

Цель задачи: выбрать язык программирования и среду разработки для реализации проекта.

Функции: инструментарий программиста для разработки ПО.

Входные данные: описание о языках программирования и средах

разработки.

Выходные данные: выбранный язык программирования и среды разработки.

Лицо, для которого ставится задача: Программист.

Задача 3. Разработка необходимой документации.

Цель задачи: Подготовить перечень документов, относящихся к

разработке данного ПО.

Функции: Информирование пользователя или заказчика о

возможностях программы и еѐ функционале.

21

Входные данные: Информация по оформлению документов согласно

ГОСТ.

Выходные данные: Готовая документация проекта.

Лицо, для которого ставится задача: программист.

Задача 4. Разработать логическую и физическую модели баз данных.

Цель задачи: Показать всевозможные связи и взаимодействия данных

между собой внутри одной базы данных.

Функции: Аналитическая схема взаимодействия процессов внутри

системы.

Входные данные: Информация для внесения в таблицы базы данных.

Выходные данные: Организационные связи между таблицами внутри

одной базы данных.

Лицо, для которого ставится задача: программист.

Задача 5. Разработать базу данных.

Цель задачи: Объединить данные в единую систему.

Функции: Добавление, чтение, удаление.

Входные данные:

 Информация о преподавателе;

 Информация о расписании;

 Информация о праздниках;

 Информация о дополнительных рабочих днях.

Выходные данные: Таблица, содержащая все данные.

Лицо, для которого ставится задача: программист.

Задача 6. Разработать интерфейс.

Цель задачи: Разработать пользовательскую оболочку для данного

проекта.

Функции: Связующее звено между программой и пользователем.

Входные данные: концепция создания и работы приложения.

Выходные данные: пользовательский интерфейс.

22

Лицо, для которого ставится задача: программист.

Задача 7. Запрограммировать необходимые компоненты программы.

Цель задачи: Реализовать отдельные компоненты для правильной

работы программы.

Функции: Выполнение надлежащих функций для каждого из модулей

программы.

Входные данные: Команды языка программирования.

Выходные данные: Готовые модули для работы программы.

Лицо, для которого ставится задача: программист.

Задача 8. Провести тестирование работоспособности программы.

Цель задачи: Произвести анализ работоспособности программы.

Функции: Поиск неисправностей на основе входных тестов программы.

Входные данные: Исходные тесты для проверки программы.

Выходные данные: Перечень найденных ошибок.

Лицо, для которого ставится задача: программист

Задача 9. Исправить возникшие ошибки.

Цель задачи: Довести продукт до правильного состояния.

Функции: Устранение неисправностей с целью лучшего обеспечения

функциональности программы.

Входные данные: Перечень найденных ошибок.

Выходные данные: исправленный продукт.

Лицо, для которого ставится задача: программист.

## Выбор инструментов, сред, языков

Выбор того или иного инструмента должен чем-то обуславливаться. Поэтому, что бы понять, как выбрать инструмент, необходимо выделить ключевые требования, по которым инструмент и будет подбираться.

Требования к продукту и его разработке:

* Кроссплатформенность. Приложение должно запускаться как на Android, так и на IOS.
* Время на разработку, отладку и тестирование. 3-4 месяца.
* Система контроля версий. Над проектом работает 3 программиста, поэтому система контроля версий необходима.

Язык программирования и технология для разработки мобильных приложений.

Был выбран язык программирования C# и технология разработки кроссплатформенных мобильных приложений Xamarin.Froms. Язык был определён сразу потому, что он хорошо знаком всем программистам в команде и потому, что платформа Xamarin.Forms работает только на C#.

Xamarin.Forms позволяет писать один код, который будет работать на всех выбранных платформах. Исключение составляют лишь платформозависимый код (работа с файлами, системные настройки и т.д.) и комплексный интерфейс. За счёт небольшого количества платформозависимого кода время на разработку сокращается.

Среда разработки

Где C# там и Visual Studio. Эта IDE, а именно Visual Studio Community 2019 является бесплатной в использовании и хорошо знакома всем программистам в команде. В этой среде очень удобно работать с различными библиотеками. Использование нового фреймворка .NET Core, который в 2020 году должен слиться с .NET Framework в единый .NET 5, позволяет избавиться от основного ограничения обычного .NET Framework - привязанность к ОС Windows. .NET Core - написанная с нуля кроссплатформенная реализация .NET. Как и обычный .NET, она была создана Microsoft, то уже в формате open-source, из-за чего развитие .NET Core идёт более активно и прозрачно. Таким образом, C# стал платформо-независимым языком программирования.

Исходя из задач были выбраны следующие технологии:

СУБД MySQL — популярная, быстрая и надёжная реляционная база данных, предоставляющая широкие возможности настройки, администрирования, разработки всех компонентов базы данных и управления ими.

Выбор её был обусловлен, наличием достаточных знаний в функционале данной СУБД, простотой её использования, а также тем что существует готовый LINQ-to-SQL провайдер для .NET Core для взаимодействия с MySQL.

Для работы с MySQL была выбрана программа DataGrip - кроссплатформенная IDE для работы с базами данных от JetBrains. Её преимущества:

1. Возможность использования в любой операционной системе;

2. Высокая производительность;

3. Широкий выбор поддерживаемых СУБД, включая MySQL;

4. Возможность подключения как к локальной БД, так и к удалённой по защищённому протоколу;

5. Инструмент разметки таблиц, включающий в себя визуальные редакторы для столбцов, индексов, первичных ключей, внешних ключей, проверки целостности, статистики, свойств хранилищ.

6. Инструмент создания диаграмм базы данных для визуализации ее структуры и построения логических отношений между таблицами;

7. Редактор SQL запросов с контекстными и интеллектуальными подсказками в процессе написания;

8. Инструменты автоматического рефакторинга таблиц;

9. Возможность экспорта всех данных в CSV, TXT и XML файлы, а также экспорт структуры базы данных и самих данных при помощи mysqldump через графический интерфейс.

Для разработки серверной части приложения использовался JetBrains Rider - кроссплатформенная IDE для .NET, поддерживающая разработку консольных, серверных и графических приложений на .NET Framework и .NET Core, мобильных приложений на Xamarin и видеоигр на Unity. Для анализа C# кода в Rider используется ReSharper - широко известное расширение для IDE Visual Studio от Microsoft собственной разработки JetBrains. Он обеспечивает более умную, адаптивную и широкую помощь разработчику, чем IntelliSense в Visual Studio. Помимо этого, Rider значительно более производительный, чем другие IDE для .NET. Другие полезные особенности Rider:

1. Широкий набор средств рефакторинга и генерации шаблонного кода, такого как реализации интерфейса или абстрактного класса, конструкторы, override методы, свойства с приватным backed полем и другие.

2. Встроенный декомпилятор;

3. Встроенный NuGet клиент;

4. Инструмент для просмотра скомпилированного IL кода;

5. Поддержка unit тестирования при помощи платформ MSTest, xUnit и NUnit;

6. Интегрированный REST клиент, позволяющий прямо в IDE выполнять заданные HTTP запросы для тестирования веб-сервера;

7. Инструмент для базового взаимодействия с базами данных через графический интерфейс;

8. Мощный отладчик кода;

9. Интеграция с системами контроля версий (VCS), позволяющая управлять коммитами, версиями, ветками и другими компонентами VCS через интерфейс;

10. Встроенный профайлер программ;

11. Богатый репозиторий плагинов, от меняющих внешний вид до значительно расширяющих функционал.

## Структура программного комплекса

АМБДК включает в себя следующие элементы:

Веб интерфейса - включающего в себя страницы регистрации\авторизации, добавления и просмотра транзакций.

BackEnd - включающая в себя контроллеры входящих запросов, базы данных и методы её изменения.

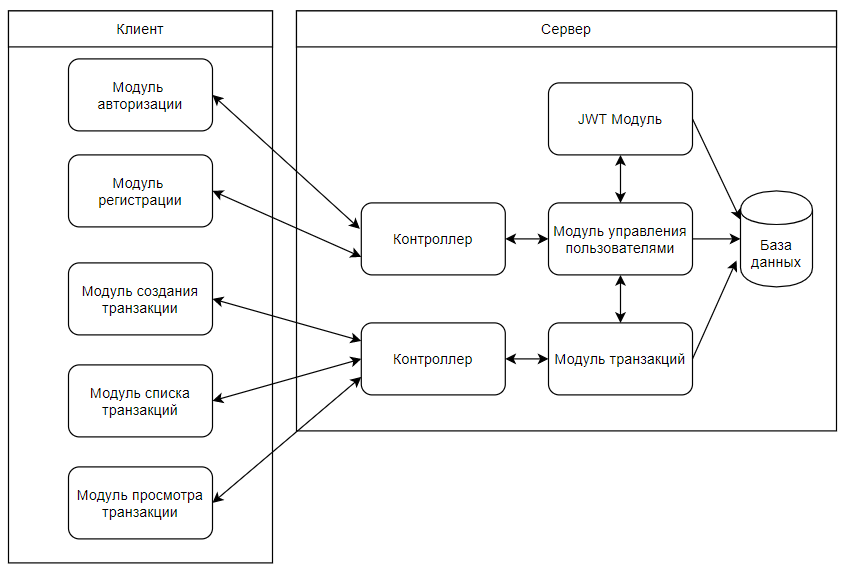


Рисунок 2.2. Схематичное отображение структуры проекта

### Логическое и физическое моделирование базы данных

Организация базы данных — это представление пользователя о той предметной области, информация о которой должна храниться в БД.

Результатом проектирования БД является набор из трёх ключевых составляющих схемы БД:

1. информация об объектах БД, относящихся к выбранной предметной области;
2. информация о свойствах этих объектов;
3. информация о связях и отношениях между этими объектами.

Такая модель не зависит от технической реализации базы данных и ЭВМ, на которой она функционирует.

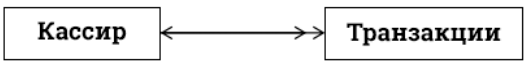
В предметной области «формирование транзакций» можно выделить следующие объекты:

объекты: Кассиры, Транзакции.

В предметной о,ласти можно определить следующие сущности:

* Сущность Кассиры атрибуты: Логин кассира, код кассира, id, пароль.
* Сущность Транзакции атрибуты: код кассира, массив мебели (номер, наименование, количество, цена за 1 ед), внесённая сумма, тип операции.

Рассмотрим типы связей между сущностями кассир и транзакции. Учитывая, что кассир видит не только свои транзакции - Связь один ко многим.



Структуру таблицы и их связей можно представить в виде ER-диаграммы:

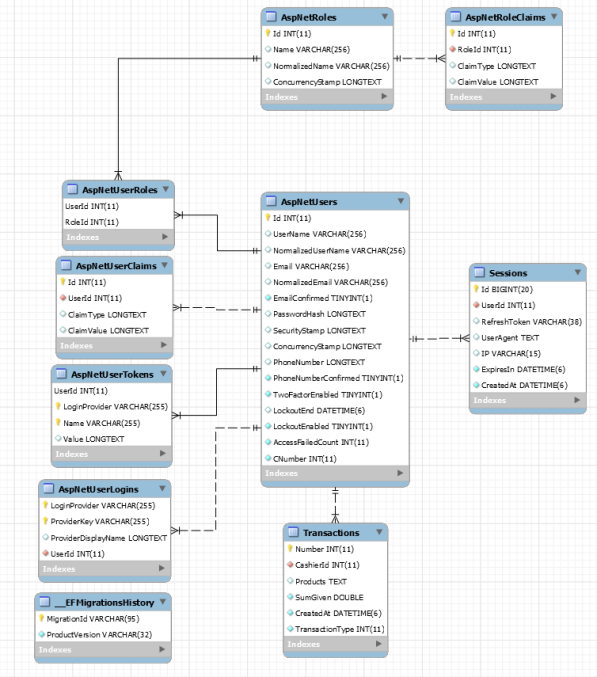


Рис. 2.3.1.*ER*-диаграмма

Структура отношения Кассир можно представить в виде таблицы:

Таблица 1 Структура отношения Кассиры

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| Код Кассира | Числовой |
| Логин | Текстовый |
| Пароль | Текстовый |

Структуру отношения Транзакции можно представить в виде таблицы 2.2.

Таблица 2 .Структура отношения Транзакции

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| Номер транзакции | Числовой |
| Дата | Время |
| Массив мебели | Массив |
| Внесённая сумма | Числовой |
| Итоговая сумма | Числовой |
| Кассир | Объект |
| Сдача | Числовой |

Таблица «Пользователи» содержит общую информацию о

пользователях/ кассирах.

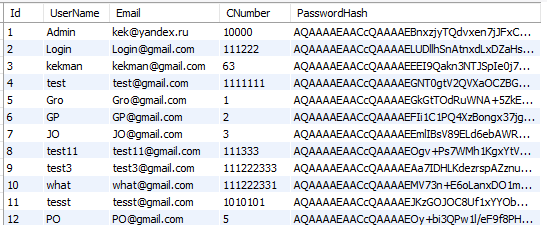


Рисунок 2.3.2. Основная часть структура таблицы “Пользователи” в режиме конструктора

Таблица «Транзакции» содержит информацию о добавленных транзакциях.

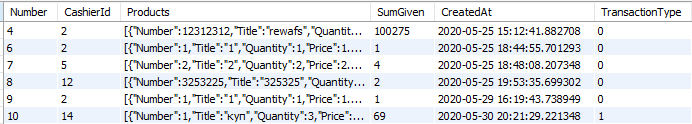


Рисунок 2.3.3 Структура таблицы «Транзакции» в режиме конструктора

Таблица «Сессии» содержит информацию о текущих авторизованных сессиях.

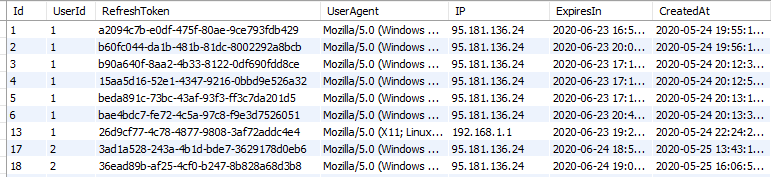


Рисунок 2.3.4 Структура таблицы «Сессии» в режиме конструктора

### Разработка автоматизированной системы

Описание структуры модулей программы:

Модуль авторизации. На рисунке 2.4.1 представлено окно авторизации:

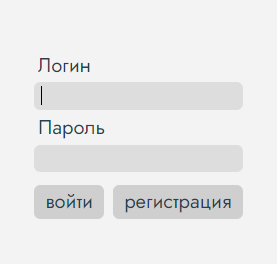


Рис. 2.4.1. Окно авторизации

Чтобы успешно подключится, необходимо заполнить все поля и нажать кнопку c изображением стрелки вправо.

Модуль регистрации

На рисунке 2.4.2 представлено окно регистрации, чтобы попасть в нее из формы регистрации, требуется нажать на кнопку «Регистрация» в форме авторизации. Далее требуется заполнить все поля и нажать на кнопку «Регистрация», чтобы в будущем была возможность авторизации в программе. Либо нажать на кнопку «Назад», чтобы вернуться к форме авторизации.

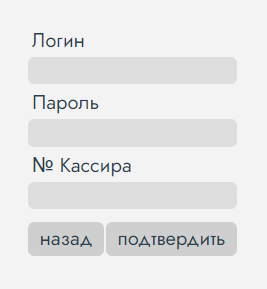


Рис.2.4.2. Окно регистрации

Модуль главного меню На рисунке 2.4.4. представлено окно главного меню, которое является связующим элементом между модулями программы. На выбор представлены три кнопки для перехода между модулями:

* Произвести операцию;
* Все отчёты;
* Выход.

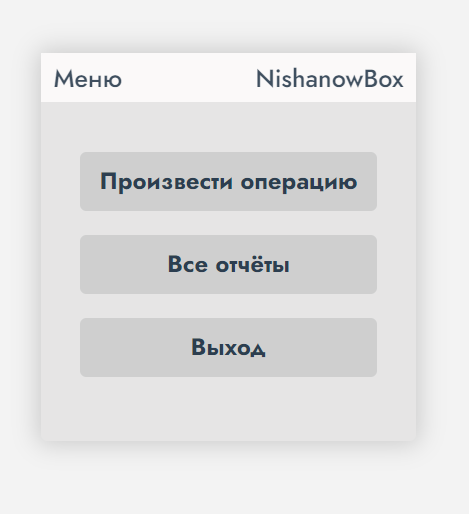


Рис. 2.4.4. Главное меню

На рисунках 2.4.4. и 2.4.5 представлены операции «Продажа» и «Возврат». Здесь можно выбрать количество мебели и набрать следующие данные: номер, название, кол-во и цену за 1ед. После того как введены следующие данные, идет вычисление суммы за товар. По окончании операции кассир может отправить информацию в отчет.

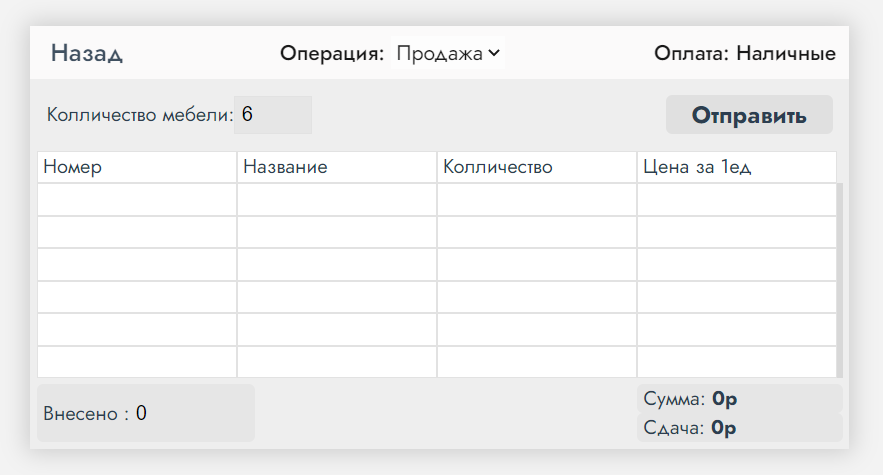


Рис. 2.4.4. Операция «Продажа»

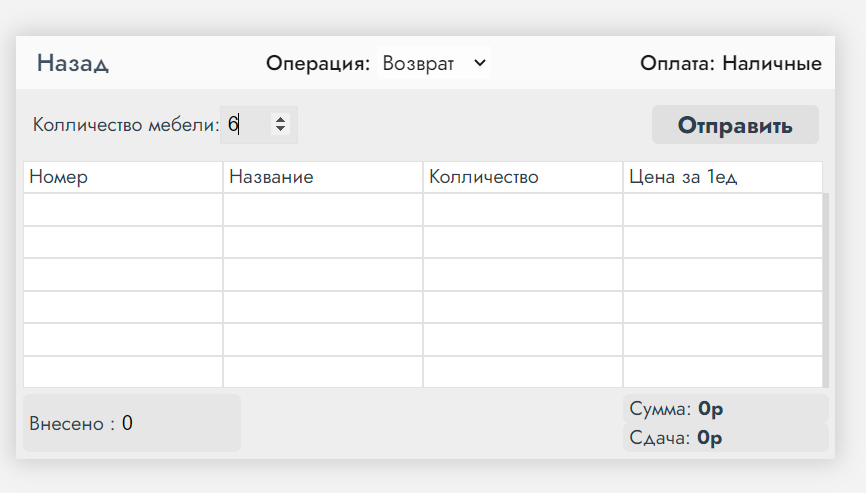


Рис. 2.4.5. Операция «Возврат»

На Рис. 2.4.6. Представлены отчеты по продажам и возвратам. Здесь есть следующие данные в таблице: № транзакции, имя, время, сумма.

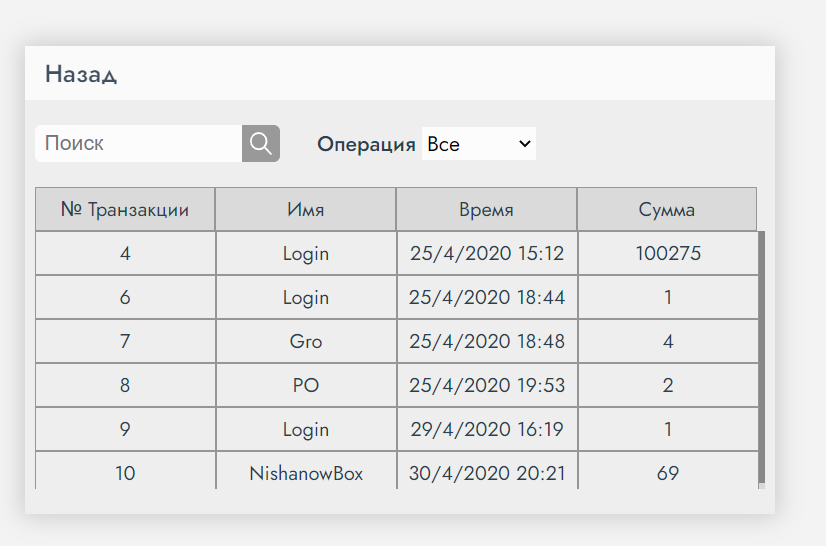


Рис. 2.4.6. Отчеты по продажам и возвратам.

На рисунке 2.4.7. Приведена одна из транзакций кассира NishanowBox. Здесь имеются сведения о названии товара, количестве, цены за 1 единицу, сам Кассир, общая стоимость товара и время добавления. Также можно распечатать эту транзакцию, нажав на кнопку печать. Пример приведен на рисунке 2.4.8.

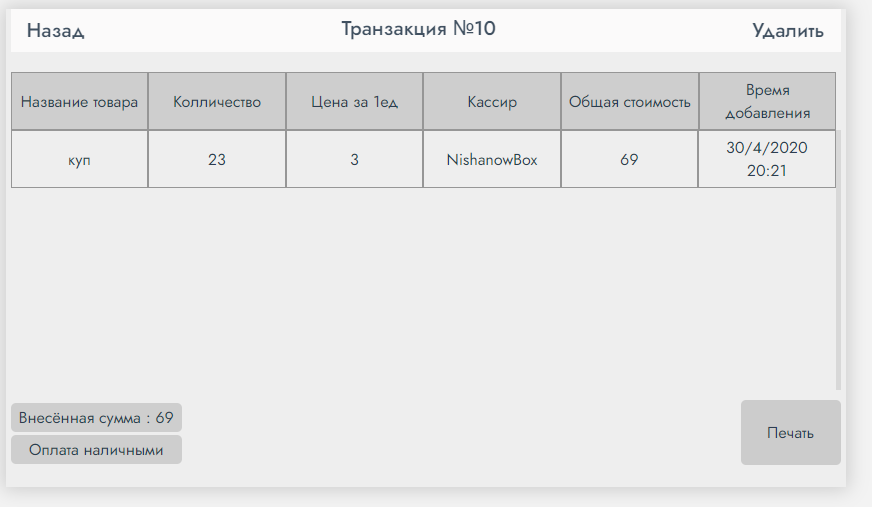


Рис.2.4.7. Транзакция №10 от NishanowBox

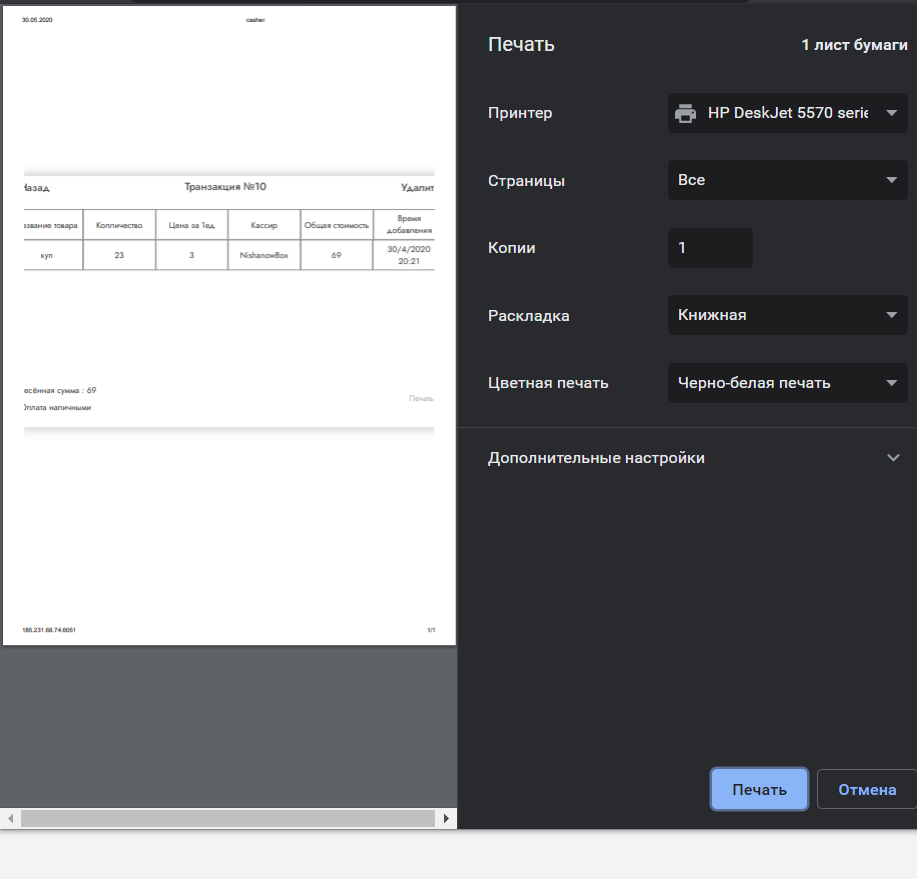


Рис.2.4.8. Печать транзакции №10 от NishanowBox.

### Описание тестовых наборов

1. **Авторизация**

**Константы:** Логин для кассира Нишанова = NishanowBox, Пароль = Nish1234

**Входные данные:** Логин, пароль.

**Тестовый сценарий работы:** Кассир авторизуется под своей учетной записью, вводя логин и пароль. Логин и пароль могут состоять только из латинских букв и цифр. В случае если кассир включит русскую раскладку или допустит ошибку в логине или пароле, то программа выдаст сообщение о неправильном вводе. А если кассир введет все правильно, то программа выдаст сообщение о успешном входе.

**Ожидаемый результат:** Сообщение об успешном входе.

**Тестовые наборы:**

1) ввожу логин: NishanowBox

Ввожу пароль: Nish1234

Результат: сообщение об успешном входе.

2) ввожу логин: 1234йцук

Ввожу пароль: qwer4321

Результат: неверный логин или пароль.

3) ввожу логин: 1234qwer

Ввожу пароль: йцук4321

Результат: неверный логин или пароль.

4) ввожу логин: 1234йцук

Ввожу пароль: йцук4321

Результат: неверный логин или пароль.

5) оставляю поле с логином или паролем пустыми

Результат: пожалуйста проверьте правильность введенных данных.

1. **Меню**

**Тестовый сценарий работы:** После того, как кассир успешно авторизовался, ему в поле выходит меню, в котором он может выбрать режим оплаты или возврата товара, а также выйти с аккаунта.

**Ожидаемый результат:** выбор между элементами меню «Оплата товара», «Возврат товара» и «Выход».

1. **Оплата товара.**

**Тестовый сценарий работы:** Открывается окно с интернет-корзиной. Кассир оформляет заказ от покупателя по штрихкоду от каждого товара и все данные о товаре (Наименование мебели, цена за 1 шт., кол-во и сумма) добавляются в интернет-корзину. Сумма высчитывается по формуле **цена \* кол-во = сумма.** Также под этим списком товаров есть еще и данные «Итого», в котором записывается суммарная цена за все товары. Кассир подсчитывает все это дело, покупатель платит и получает квитанцию.

**Ожидаемый результат:** Подсчет суммы и итого по формуле и вывод этих данных на экране.

**Тестовый набор:**

Наименование товара: стул

Цена: 20000

Кол-во: 4

Сумма = цена \* кол-во = 80000

1. **Возврат товара.**

**Тестовый сценарий работы:** Кассир оформляет возврат товара от покупателя. Если есть чек и штрихкод на мебели остался и не истек срок гарантии, то покупатель может вернуть товар, в противном случае – нет. Кассир пробивает чек и штрихкод на подлинность. В случае если чек и штрихкод действителен, то покупатель может вернуть товар и получить за него обратно деньги, в противном случае нет.

**Ожидаемый результат:** Возврат товара и цены за него.

1. **Выход.**

**Тестовый сценарий работы:** Кассир может просто выйти с аккаунта, нажав на кнопку выход и подтвердив свой выбор.

**Ожидаемый результат:** Выход из программы.

## Описание процедур и функций

Данная программа состоит из 10 основных форм, класса для перемещения данных между формами и класса, который отвечает за запуск программного продукта. Каждая форма имеет функции и методы для полноценной работы программы. Далее рассмотрим формы и их методы и функции:

* AuthPage – Имеет 2 режима - страница входа\ страница регистрации. На них расположены поля ввода логина и пароля в случае авторизации, и поля ввода логина, пароля и кода кассира в случае регистрации соответственно. Имеет методы отсылки данных на сервер.
* MainPage - это окно главного меню. Имеет только методы, вызываемые элементами формы;
* AddPage– страница для добавления транзакций. Содержит в себе таблицу для заполнения данных о мебели. А также следующие функции:

1. SetFurnutureMass - Формирование массива мебели необходимого размера
2. SetType – Выбор режима транзакции;
3. PostDate – Отправка данных на сервер;

ReportsPage - Страница просмотра транзакций с последующим выбором конкретной транзакции для более подробного изучения. Включает в себя следующие функции:

1. GetReports - GET запрос на сервер для получения списка транзакций.
2. ParseDate - метод формирования даты и времени конкретной транзакции.
3. SearchMehod - метод поиска по транзакциям (номер, цена, имя)
4. ShowModal - метод открывающий окно с дополнительной информацией по выбранной транзакции.
5. DeleteTrasiction - DELETE запрос на сервер для удаления транзакции.
6. Print - метод для вызова окна печати отчёта.

Серверная часть построена по архитектуре MVC (Model-View-Controller) и состоит из 3-х контроллеров:

1. AppController - отвечает за отдачу клиентского SPA приложения в ответ на на любой запрос, URL которого не удалось соотнести с каким-либо другим контроллером.
2. UsersController - предоставляет пути для всех действий, связанных с пользователями. В частности:
   1. POST /api/users - путь для добавления нового пользователя;
   2. DELETE /api/users/{number} - путь для удаления пользователя с указанным номером;
   3. POST /api/users/auth - путь для авторизации пользователя в системе;
   4. POST /api/users/refresh - путь для обновления пары JWT токенов.
3. TransactionsController - обеспечивает выполнение операций, связанных с транзакциями. В частности:
   1. POST /api/transactions - путь для добавления новой транзакции;
   2. GET /api/transactions - путь для получения списка всех транзакций;
   3. DELETE /api/transactions/{number} - путь для удаления транзакции с указанным номером.

При этом все пути TransactionsController-а требуют, чтобы пользователь авторизован. В ином случае пользователь получит ошибку HTTP 401.

Описанные контроллеры не содержат в себе логики непосредственной манипуляции с базой данных. При помощи инъекции зависимостей (DI) они получают доступ к сервисам, которые уже выполняют операции над моделями. Есть три отдельных сервиса:

1. JwtAuthService - отвечает за формирование и обновление JWT токенов и хранение в СУБД списка активных сессий. Каждая сессия состоит из refresh token-а, IP пользователя, User-Agent-а пользователя и срока годности refresh token-а. Хранение такого набора данных позволяет по необходимости отзывать refresh token-ы, а также исключить вероятность получения доступа к защищённым частям приложения третьими лицами при помощи перехвата access token-а. JwtAuthService не зависит от сторонних элементов базы данных и структуры модели пользователя.
2. UsersService - предоставляет необходимый для UsersController функционал. Для авторизации пользователя и обновления пары токенов использует JwtAuthService, который получает через DI.
3. TransactionsService - сервис для управления транзакциями. Для связи транзакции с пользователем используется UsersService, также получаемый через DI.