СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 7](#_Toc122433002)

[1 Постановка задачи 8](#_Toc122433003)

[1.1Анализ предметной области 8](#_Toc122433004)

[1.2 Техническое задание на разработку ГОСТ 19.201-78 9](#_Toc122433005)

[1.2.1 Введение 9](#_Toc122433006)

[1.2.1.1 Наименование программы 9](#_Toc122433007)

[1.2.1.2 Краткая характеристика области применения 9](#_Toc122433008)

[1.2.2 Основания для разработки 9](#_Toc122433009)

[1.2.2.1 Основание для проведения разработки 9](#_Toc122433010)

[1.2.3 Назначение разработки 9](#_Toc122433011)

[1.2.3.1 Функциональное назначение 9](#_Toc122433012)

[1.2.3.2 Эксплуатационное назначение 9](#_Toc122433013)

[1.2.4 Требования к программе или программному изделию 9](#_Toc122433014)

[1.2.4.1 Требования к функциональным характеристикам 9](#_Toc122433015)

[1.2.4.2 Требования к надежности 10](#_Toc122433016)

[1.2.4.3 Время восстановления после отказа 10](#_Toc122433017)

[1.2.4.4 Отказы из-за некорректных действий оператора 10](#_Toc122433018)

[1.2.4.5 Условия эксплуатации 10](#_Toc122433019)

[1.2.4.6 Требования к видам обслуживания 10](#_Toc122433020)

[1.2.4.7 Требования к численности и квалификации персонала 10](#_Toc122433021)

[1.2.4.8 Требования к составу и параметрам технических средств 11](#_Toc122433022)

[1.2.4.9 Требования к информационной и программной совместимости 11](#_Toc122433023)

[1.2.4.10. Требования к маркировке и упаковке 11](#_Toc122433024)

[1.2.4.11 Требования к программной документации 12](#_Toc122433025)

[1.2.5 Технико-экономические показатели 12](#_Toc122433026)

[1.2.6 Стадии и этапы разработки 12](#_Toc122433027)

[1.2.6.1 Стадии разработки 12](#_Toc122433028)

[1.2.6.2 Этапы разработки 12](#_Toc122433029)

[1.2.6.3 Содержание работ по этапам 13](#_Toc122433030)

[1.2.7 Порядок контроля и приемки 13](#_Toc122433031)

[2. Проектирование 14](#_Toc122433032)

[2.1 Функциональная структура программы 14](#_Toc122433033)

[2.2 Схема модулей программы 14](#_Toc122433034)

[2.3Паспорта основных модулей 15](#_Toc122433035)

[2.3.1 Модуль «Добавление товара в чек» 15](#_Toc122433036)

[2.3.2 Модуль «Проверка вводимых данных» 17](#_Toc122433037)

[2.3.3 Модуль «Поиска товара по названию» 18](#_Toc122433038)

[2.3.4 Модуль «Вывод чека» 19](#_Toc122433039)

[3 Руководство оператора ГОСТ 19.505-79 20](#_Toc122433040)

[3.1 Назначение программы 20](#_Toc122433041)

[3.1.1 Функциональное назначение 20](#_Toc122433042)

[3.1.2 Состав функций 20](#_Toc122433043)

[3.2 Условия выполнения программы 20](#_Toc122433044)

[3.2.1 Климатические условия эксплуатации 20](#_Toc122433045)

[3.2.2 Требования к составу и параметрам технических средств 20](#_Toc122433046)

[3.2.3 Требования к программным средствам, используемым программой 20](#_Toc122433047)

[3.3 Требования к персоналу (пользователю) 21](#_Toc122433048)

[3.4 Выполнение программы 21](#_Toc122433049)

[3.4.1 Загрузка и запуск программы 21](#_Toc122433050)

[3.4.2 Выполнение программы 22](#_Toc122433051)

[3.4.2.1 Функция заполнения начальными данными 22](#_Toc122433052)

[3.4.2.2 Функция добавления товара в чек 22](#_Toc122433053)

[3.4.2.3 Функция показать чек 24](#_Toc122433054)

[3.4.2.4 Функция сброс чека 26](#_Toc122433055)

[3.4.2.5 Функция поиска 26](#_Toc122433056)

[3.5 Сообщения оператору 28](#_Toc122433057)

[3.5.1 Сообщение об ошибки введении некорректных данных 28](#_Toc122433058)

[3.5.1 Сообщение превышение количества товара 28](#_Toc122433059)

[4 Тестирование программы 29](#_Toc122433060)

[4.1 Программа, методика и результаты испытаний ГОСТ 19.301-79 29](#_Toc122433061)

[4.1.1 Объект испытаний 29](#_Toc122433062)

[4.1.1.1 Наименование программы 29](#_Toc122433063)

[4.1.1.2 Назначение разработки 29](#_Toc122433064)

[4.1.2 Цель испытаний 29](#_Toc122433065)

[4.1.2.1 Основания для проведения испытаний 29](#_Toc122433066)

[4.1.2.2 Место и продолжительность испытаний 29](#_Toc122433067)

[4.1.2.3 Перечень документов, предъявляемых на испытания 29](#_Toc122433068)

[4.1.2.4 Объем испытаний 29](#_Toc122433069)

[4.1.2.4.1 Перечень проверок, проводимых на 1 этапе испытаний 30](#_Toc122433070)

[4.1.2.4.2 Перечень проверок, проводимых на 2 этапе испытаний 30](#_Toc122433071)

[4.1.2.4.3 Количественные и качественные характеристики, подлежащие оценке 30](#_Toc122433072)

[4.1.3 Требования к программе 30](#_Toc122433073)

[4.1.4 Требования к программной документации 30](#_Toc122433074)

[4.1.5 Средства и порядок испытаний 31](#_Toc122433075)

[4.1.5.1 Технические средства, используемые во время испытаний 31](#_Toc122433076)

[4.1.5.1.1 Порядок проведения испытаний 31](#_Toc122433077)

[4.1.5.2 Условия и порядок проведения испытаний 31](#_Toc122433078)

[4.1.5.2.1 Условия проведения испытаний 31](#_Toc122433079)

[4.1.5.2.2 Условия начала и завершения отдельных этапов испытаний 31](#_Toc122433080)

[4.1.5.2.3 Требования к техническому обслуживанию 32](#_Toc122433081)

[4.1.6. Методика испытаний 32](#_Toc122433082)

[4.2 Протокол испытаний 33](#_Toc122433083)

[4.2.1 Модуль «Добавления товара в чек» 33](#_Toc122433084)

[4.2.2 Модуль «Вывод чека» 35](#_Toc122433085)

[ЗАКЛЮЧЕНИЕ 37](#_Toc122433086)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 38](#_Toc122433087)

[ПРИЛОЖЕНИЕ А 39](#_Toc122433088)

[ПРИЛОЖЕНИЕ В 41](#_Toc122433089)

# ВВЕДЕНИЕ

В современных условиях, человек вынужден работать с гигантскими объемами информации. Системы автоматизированного учета обязаны представлять собой мощные средства, способные обрабатывать огромные потоки данных высокой структурной сложности за минимум затраченного времени, обеспечивая дружественный диалог с пользователем

Цель данного проекта состоит в проектировании, реализации и тестировании программного продукта, реализующего работу с информационной системой.

Данные будут хранится в виде таблиц. Для взаимодействия с ними будет применяться СУБДКафе-столоваяи среда разработки Visual Studio на языке программирования С#.

Для того чтобы достичь цели курсового проекта необходимо поставить перед собой следующие задачи:

* Более углубленно изучить язык программирования С#;
* Изучить СУБД Колледж;
* Изучить набор данных в Microsoft Visual Studio 2022;
* Изучить нормализацию БД;
* Изучить и проанализировать соответствующую литературу.

## 1 Постановка задачи

## Анализ предметной области

Проект будет реализовывать программный продукт для автоматизации работы с данными Кафе-столовой. Будет использоваться удаленный сервер для хранения данных и клиентское приложение. Интерфейс должен быть удобен и интуитивно понятен для пользователя с минимальным опытом знакомства с подобными системами.

В БД должны храниться сущности и атрибуты данной темы:

* Сущность Tovar содержит атрибуты: *Id* – является ключом, Name, Units, KolvoOst, PriceOne;

Разработка подобной системы актуальна на данный момент, так как она автоматизирует работу кассиров и других пользователей. Она обеспечивает автоматизацию в расчете скидок на определенные товары и на определенные суммы в чеке. Вывод всей нужной информации о товаре, необходимой для покупателя.

## 1.2 Техническое задание на разработку ГОСТ 19.201-78

### 1.2.1 Введение

### 1.2.1.1 Наименование программы

Наименование программы – «Кафе-столовая».

#### 1.2.1.2 Краткая характеристика области применения

Программа предназначена для просмотра информации товарах кафе-столовой, а также для хранение этой информации.

### 1.2.2 Основания для разработки

#### 1.2.2.1 Основание для проведения разработки

Основанием для проведения разработки является задание по учебной практике.

### 1.2.3 Назначение разработки

#### 1.2.3.1 Функциональное назначение

Функциональным назначением программы является работа с данным, корректный вывод данных.

#### 1.2.3.2 Эксплуатационное назначение

Не имеет. Программа предназначена для решения элементарных задач обработки данных.

### 1.2.4 Требования к программе или программному изделию

#### 1.2.4.1 Требования к функциональным характеристикам

Программа должна обеспечивать возможность выполнения следующих функций:

* добавления товара в чек
* очистка чека
* Поиск товара по названию
* Вывод общей суммы с учетом скидок
* Очистка поиска

#### 1.2.4.2 Требования к надежности

Требования к обеспечению надежного функционирования программы не предъявляются.

#### 1.2.4.3 Время восстановления после отказа

Время восстановления после не фатального сбоя ОС не должно превышать 10-ти минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после фатального сбоя не должно превышать времени, требуемого для устранения неисправностей технических средств и переустановки ПО.

#### 1.2.4.4 Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой. Во избежание возникновения отказов программы по указанной выше причине следует обеспечить работу пользователя без предоставления ему административных привилегий.

#### 1.2.4.5 Условия эксплуатации

Климатические условия эксплуатации, при которых должны осуществляться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам (ПК) в части условий эксплуатации.

#### 1.2.4.6 Требования к видам обслуживания

Программа не требует проведения каких-либо видов обслуживания.

#### 1.2.4.7 Требования к численности и квалификации персонала

Минимальное количество персонала должно составлять 2 штатных единиц - системный администратор и пользователь ПК – оператор.

Системный администратор должен иметь высшее профильное образование. В перечень задач системного администратора входят:

* поддержание работоспособности технических средств;
* установка и поддержка работоспособности системных программных средств – ОС;
* установка программ.

Оператор должен уметь работать с графическим пользовательским интерфейсом ОС.

#### 1.2.4.8 Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер, включающий в себя:

* Процессор: 1,8 ГГц и более;
* Свободное место на жестком диске: 850Мб и более;
* ОЗУ: 4 ГБ и более
* Видеоадаптер: WXGA (1366 на 768), рекомендуется (1920 на 1080)

#### 1.2.4.9 Требования к информационной и программной совместимости

**Требования к информационным структурам и методам решения**

Требования к информационным структурам (записям) на входе, а также к методам решения не предъявляются. Информационная структура должна включать отображение входных данных и результата на экране.

**Требования к исходным кодам и языкам программирования**

Программа реализована на языке С#. Для разработки приложения и базы данных были использованы среды Microsoft Visual Studio 2022 и MS SQL Server 2019.

**Требования к защите информации и программ**

Требования к защите информации и программ не предъявляются.

**Специальные требования**

Программа должна обеспечивать взаимодействие с пользователем посредством графического интерфейса.

#### 1.2.4.10. Требования к маркировке и упаковке

**Требование к маркировке**

Программное изделие [оптический диск] должно иметь маркировку с обозначением фамилии студента, номера группы, наименования программы. Маркировка должна быть нанесена на программное изделие в виде надписи, маркером на диск или ручкой на упаковку диска.

**Требования к упаковке**

Упаковка программного изделия должна осуществляться в конверте для дисков.

#### 1.2.4.11 Требования к программной документации

В состав сопровождающей документации должны входить:

* Техническое задание на разработку ГОСТ 19.201-78
* Руководство оператора ГОСТ 19.505-79
* Программа, методика и результаты испытаний ГОСТ 19.301-79

### 1.2.5 Технико-экономические показатели

**Экономические преимущества разработки**

Коммерческое использование не предусмотрено.

### 1.2.6 Стадии и этапы разработки

#### 1.2.6.1 Стадии разработки

Разработка должна быть проведена в две стадии:

* Разработка технического задания;
* Разработка технического проекта.

#### 1.2.6.2 Этапы разработки

Этапы разработки:

* Разработка технического задания
* разработка программы;
* разработка программной документации;
* испытания программы.

#### **1.2.6.3 Содержание работ по этапам**

На этапе разработки технического задания должны быть выполнены следующие работы:

* Постановка задачи
* Определение и уточнение требований к техническим средствам
* Определение требований к программе
* Определение стадий, этапов и сроков разработки программы и документации на неё
* Выбор языка программирования и среды разработки
* Согласование и утверждение технического задания

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ.

На этапе испытаний программы долины быть выполнены следующие виды работ:

* Разработка, согласование и утверждение программы и методики испытаний
* Проведение приемосдаточных испытаний
* Корректировка программы и программной документации по результатам испытаний

На этапе подготовки и передачи программы должна быть выполнена работа по сдаче программы и программной документации преподавателю.

### 1.2.7 Порядок контроля и приемки

Приемосдаточные испытания должны проводиться в колледже преподавателем до декабря 2022 года.

Приемосдаточные испытания программы должны проводиться согласно разработанной студентом и согласованной преподавателем «Программы и методики испытаний».

# 2 Проектирование

## 2.1 Функциональная структура программы

Описываются функции, которые будет выполнять программа. Согласно технологии структурного программирования, любая составная функция может, и должна быть разложена на элементарные для данного уровня функции.

Ниже приведена таблица, в которую включаются все возможные функции, которые планируются использовать в программе в соответствии с таблицей 2.1.

**Таблица 1** – Таблица спецификации функций

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название** | **Действие** | | **Объект** | **Вид** |
| **Основные действия с БД** | | | | |
| Поиск | Поиск элементов таблице и запись в таблицу | | Таблица | Основная |
| Загрузка данных из БД | Осуществляет загрузку из БД | | Таблица | Основная |
| Добавление товара в чек | Осуществляет добавление в чек | | Таблица | Основная |
| **Вспомогательные функции** | | | | |
| Сброс чека | | Изменение данных в чеке | Таблицы | Основная |
| Показать чек | | Осуществляет показ чека | Таблицы | Основная |

## 2.2 Схема модулей программы



**Рисунок 1** – Схема модулей программы

## Паспорта основных модулей

### 2.3.1 Модуль «Добавление товара в чек»

**Таблица 2** -Структура данных функции «Добавления товара в чек»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void AddChek\_Click(object sender, RoutedEventArgs e) | Проверяет корректность введенных данных и добавляет товар в чек | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 2** – Общий алгоритм функции «Добавления товара в чек»

### 2.3.2 Модуль «Проверка вводимых данных»

**Таблица 3** -Структура данных функции «Проверка вводимых данных»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void KolvoChek\_TextChanged(object sender, TextChangedEventArgs e) | Проверяет корректность введенных данных | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 3** – Общий алгоритм функции «Проверка вводимых данных»

### 2.3.3 Модуль «Поиска товара по названию»

**Таблица 4** -Структура данных функции «Поиск товара по названию»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void FindTovar\_Click (object sender, RoutedEventArgs e) | Поиск информации о товаре по его названию | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 4** – Общий алгоритм функции «Поиск товара по названию»

### 2.3.4 Модуль «Вывод чека»

**Таблица 5** -Структура данных функции «Вывод чека»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void Window\_Loaded (object sender, RoutedEventArgs e) | Загрузка данных на форму при открытии окна | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 5** – Общий алгоритм функции «Вывод чека»

# 3 Руководство оператора ГОСТ 19.505-79

## 3.1 Назначение программы

## 3.1.1 Функциональное назначение

Функциональным назначением программы является работа с данным, корректный вывод и изменения данных.

**3.1.2 Состав функций**

Программа обеспечивает возможность выполнения перечисленных ниже функций:

* добавления товара в чек
* очистка чека
* Поиск товара по названию
* Вывод общей суммы с учетом скидок
* Очистка поиска

**3.2 Условия выполнения программы**

**3.2.1 Климатические условия эксплуатации**

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

### 3.2.2 Требования к составу и параметрам технических средств

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

* Процессор: 1,8 ГГц и более;
* Свободное место на жестком диске: 850Мб и более;
* ОЗУ: 4 ГБ и более
* Видеоадаптер: WXGA (1366 на 768), рекомендуется (1920 на 1080)

**3.2.3 Требования к программным средствам, используемым программой**

Системные программные средства, используемые программы, должны быть представлены лицензионной локализованной версией операционной системы Windows 7 и выше.

**3.3 Требования к персоналу (пользователю)**

Минимальное количество персонала, требуемого для работы программы, должно составлять не менее 2 штатных единиц – системный администратор и пользователь программы – оператор.

Системный администратор должен иметь высшее профильное образование и сертификаты компании-производителя операционной системы. В перечень задач, выполняемых системным администратором, должны входить:

* задача поддержания работоспособности технических средств;
* задачи установки (инсталляции) и поддержания работоспособности системных программных средств – операционной системы;
* задача установки (инсталляции) программы.

Пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

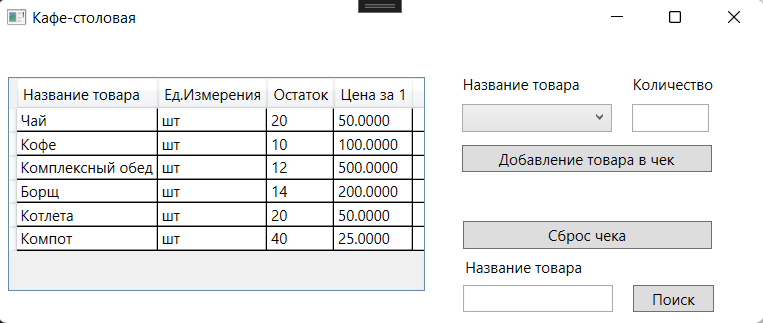
**3.4 Выполнение программы**

* Загрузка и запуск программы;
* Выполнение программы;
* Завершение работы программы.

**3.4.1 Загрузка и запуск программы**

Загрузка и запуск программы осуществляется способами, детальные сведения о которых изложены в руководстве пользователя операционной системы.

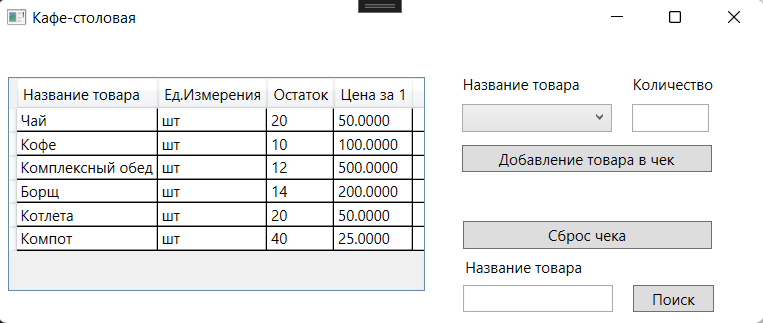
В случае успешного запуска программы на рабочем столе будет отображено главное окно приложения в соответствии с рисунком 6.

**Рисунок 6** – Главное окно приложения.

**3.4.2 Выполнение программы**

**3.4.2.1 Функция заполнения начальными данными**

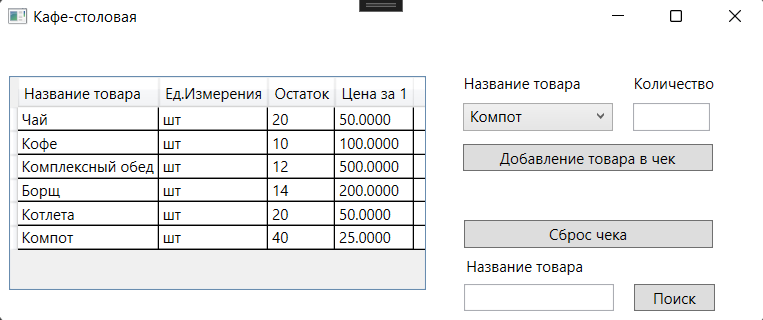
Начальные данных загружается в процессе открытия приложения из базы данных, соединенной с программой. Как показано на рисунке 7.

****

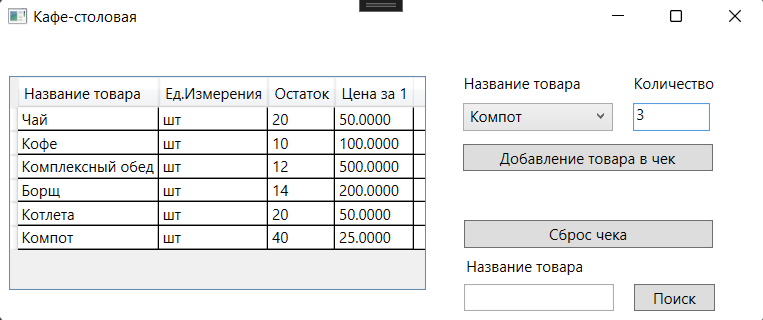
**Рисунок 7** – Выбор вкладки

**3.4.2.2 Функция добавления товара в чек**

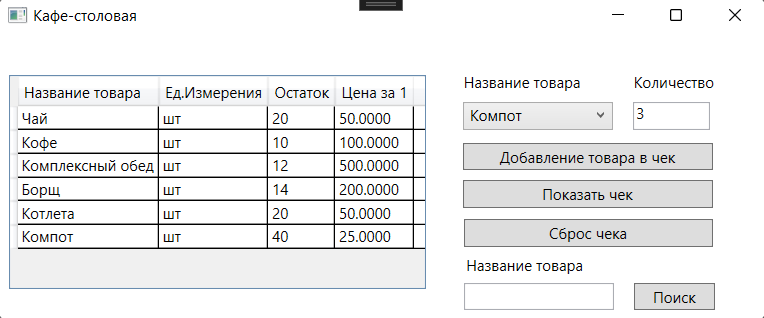
Для добавления товара в чек необходимо выбрать из выпадающего списка название товара (рисунок 8) и ввести количество товаров (рисунок 9). После заполнения всех выше, перечисленных полей нажимаем кнопку «Добавление товара в чек», после появится кнопка «Показать чек» (рисунок 10).



**Рисунок 8** – Выбор названия товара



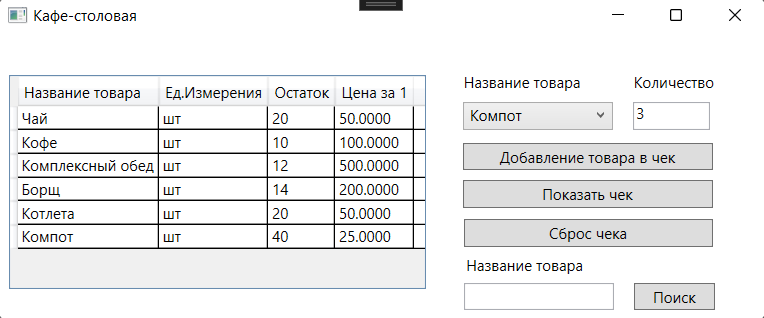
**Рисунок 9** – Ввод количества товара



**Рисунок 10** – Ввод количества товара

**3.4.2.3 Функция показать чек**

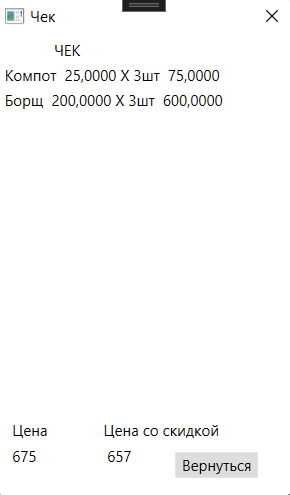
Для того, чтобы увидеть чек, необходимо добавить хотя бы один товар в чек. После появления кнопки «Показать чек», надо на неё нажать и на экран будет выведен чек. Выйти из него можно по кнопке «Вернуться». В чеке отображается: товары и их информация, сумма без учета скидки и сумма с учетом скидки если она есть.



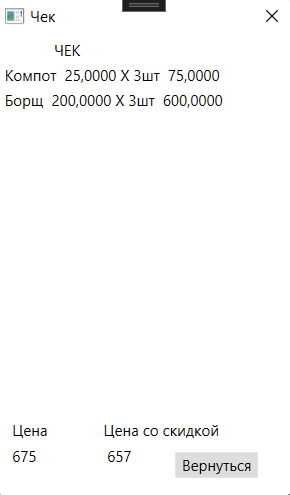
**Рисунок 11** – До нажатия на кнопку «Показать чек»



**Рисунок 12** – Чек (без учета скидки)



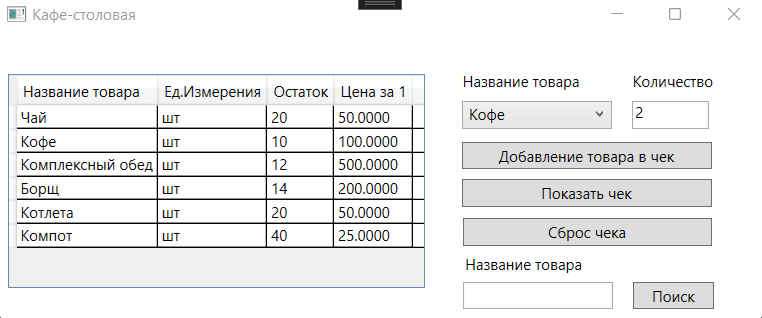
**Рисунок 13** – Чек (с учетом скидки)



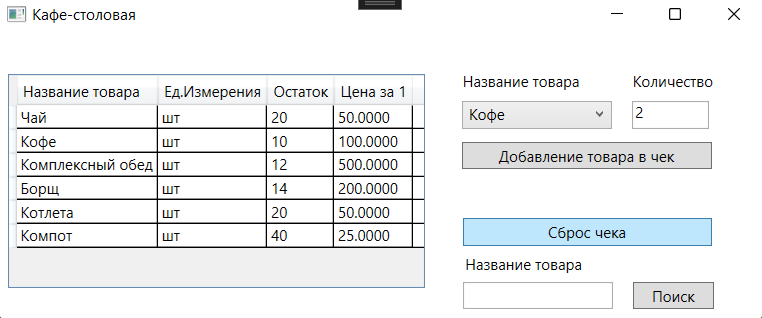
**Рисунок 14** – Кнопка «Вернуться»

**3.4.2.4 Функция сброс чека**

Для очистки чека необходимо, нажать на кнопку «Сброс чека» (рисунок 15). После не будет возможности показать чек. (рисунок 16)



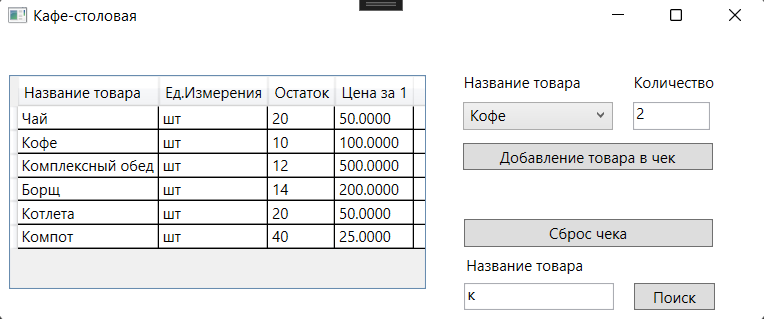
**Рисунок 15** – Кнопка «Сброс чека»



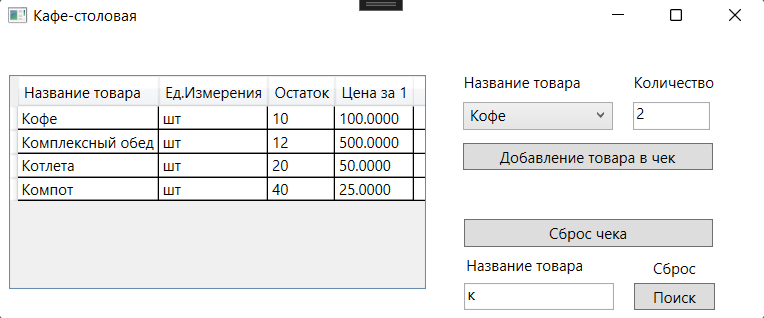
**Рисунок 16** – Программа после выполнения сброса

**3.4.2.5 Функция поиска**

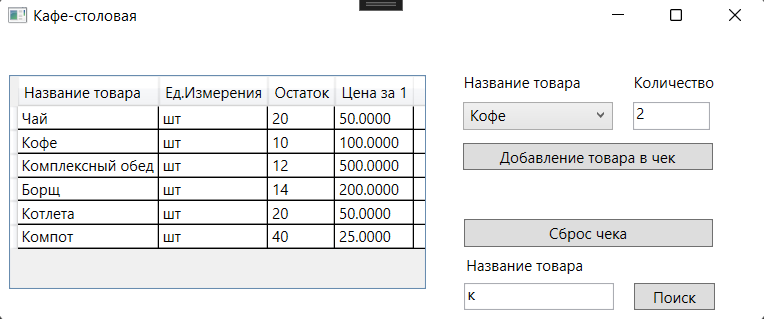
Для выполнения функции поиска необходимо заполнить поле ввода, как показано на картинке 17. Вводить данные можно как целиком слово, так и только первую букву слова. При нажатии кнопки «Поиск», найденная информация выводится, как показано на картинке 18. Для возвращения таблицы в исходное состояние, необходимо нажать кнопку «Сброс»



**Рисунок 17** – Заполнение поля ввода и кнопка «Поиск»



**Рисунок 18** - Результат поиска

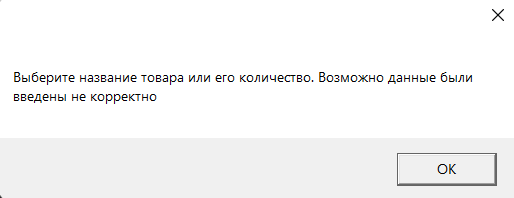


**Рисунок 19** – Исходная программа после сброса поиска

**3.5 Сообщения оператору**

**3.5.1 Сообщение об ошибки введении некорректных данных**

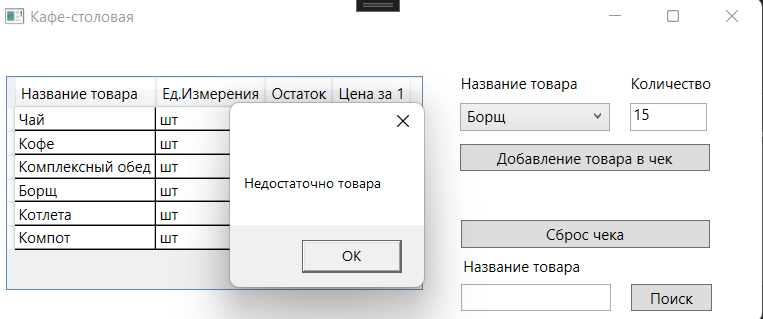
Это сообщение появится, при введении некорректных данных (рисунок 20).



**Рисунок 20** – Ошибка ввода данных.

**3.5.1 Сообщение превышение количества товара**

Это сообщение появится, при введении большего значения, чем есть на складе товара, как на рисунок 21.



**Рисунок 21** – Сообщение о недостатке товара

**4 Тестирование программы**

## 4.1 Программа, методика и результаты испытаний ГОСТ 19.301-79

### 4.1.1 Объект испытаний

#### 4.1.1.1 Наименование программы

Наименование программы – «Кафе-столовая».

#### 4.1.1.2 Назначение разработки

Функциональным назначением программы является работа с данными и корректное вывод их.

### 4.1.2 Цель испытаний

Цель проведения испытаний - проверка соответствия характеристик разработанной программы (программного изделия) функциональным и отдельным иным видам требований, изложенным в документе Техническое задание.

#### 4.1.2.1 Основания для проведения испытаний

Основанием для проведения разработки является задание по учебной практике.

#### 4.1.2.2 Место и продолжительность испытаний

Дом, Колледж. Срок 2 неделе.

#### 4.1.2.3 Перечень документов, предъявляемых на испытания

Состав программной документации должен включать в себя:

* Техническое задание на разработку ГОСТ 19.201-78
* Руководство оператора ГОСТ 19.505-79
* Программа, методика и результаты испытаний ГОСТ 19.301-79

#### 4.1.2.4 Объем испытаний

**Перечень этапов испытаний**

Испытания проводятся в два этапа:

* ознакомительный;
* испытания.

##### 4.1.2.4.1 Перечень проверок, проводимых на 1 этапе испытаний

Перечень проверок, проводимых на 1-м этапе испытаний, должен включать в себя:

* проверку комплектности программной документации;
* проверку комплектности состава технических и программных средств.

##### 4.1.2.4.2 Перечень проверок, проводимых на 2 этапе испытаний

Перечень проверок, проводимых на 2-м этапе испытаний, должен включать в себя:

* проверку соответствия технических характеристик программы;
* проверку степени выполнения требований функционального назначения программы.

##### 4.1.2.4.3 Количественные и качественные характеристики, подлежащие оценке

В ходе проведения приемосдаточных испытаний оценке подлежат качественные (функциональные) характеристики программы. Проверке подлежит возможность выполнения программой перечисленных ниже функций:

* добавления товара в чек
* очистка чека
* Поиск товара по названию
* Вывод общей суммы с учетом скидок
* Очистка поиска

### 4.1.3 Требования к программе

При проведении испытаний функциональные характеристики (возможности) программы подлежат проверке на соответствие требованиям, изложенным в п. «Требования к функциональным выполняемых функций» технического задания.

### 4.1.4 Требования к программной документации

В состав сопровождающей документации должны входить:

* Техническое задание на разработку ГОСТ 19.201-78
* Руководство оператора ГОСТ 19.505-79
* Программа, методика и результаты испытаний ГОСТ 19.301-79

### 4.1.5 Средства и порядок испытаний

#### 4.1.5.1 Технические средства, используемые во время испытаний

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ), включающий в себя:

* Процессор: 1,8 ГГц и более;
* Свободное место на жестком диске: 850Мб и более;
* ОЗУ: 4 ГБ и более
* Видеоадаптер: WXGA (1366 на 768), рекомендуется (1920 на 1080)

##### 4.1.5.1.1 Порядок проведения испытаний

Испытания должны проводиться поэтапно согласно п. «Перечень этапов испытаний» настоящего документа.

##### 4.1.5.2 Условия и порядок проведения испытаний

##### 4.1.5.2.1 Условия проведения испытаний

Испытания должны проводиться в нормальных климатических условиях по ГОСТ 22261-94. Условия проведения испытаний приведены ниже:

* температура окружающего воздуха, °С - 20 ± 5;
* относительная влажность, % - от 30 до 80;
* атмосферное давление, кПа - от 84 до 106;
* частота питающей электросети, Гц - 50 ± 0,5;
* напряжение питающей сети переменного тока, В - 220 ± 4,4.

##### 4.1.5.2.2 Условия начала и завершения отдельных этапов испытаний

Необходимым и достаточным условием завершения 1 этапа испытаний и начала 2 этапа испытаний является успешное завершение проверок, проводимых на 1 этапе (см. п. «Перечень проверок, проводимых на 1 этапе испытаний»).

Условием завершения 2 этапа испытаний является успешное завершение проверок, проводимых на 2 этапе испытаний.

##### 4.1.5.2.3 Требования к техническому обслуживанию

Требования к техническому обслуживанию не предъявляются.

**4.1.6. Методика испытаний**

**Методы проведения проверки комплектности программной документации**

Проверка комплектности программной документации на программное изделие производится визуально преподавателем. В ходе проверки сопоставляется состав и комплектность программной документации, представленной исполнителем, с перечнем программной документации.

Проверка считается завершенной в случае соответствия состава и комплектности программной документации, представленной исполнителем, перечню программной документации, приведенному в указанном выше пункте.

**Методы проведения проверки степени** **выполнения требований функционального назначения программы**

Проверка выполнения требований функционального назначения производится методом тестирования программного продукта. Тестирование будет проводится методом белого ящика по критерию C1 или условия критерия тестирования ветвей. Набор тестов в совокупности должен обеспечить прохождение каждого критерия не менее одного раза. Но применяется при тестировании ветвей. Это достаточно экономичный критерий, поскольку множество ветвей в тестируемом приложении конечно и не так велико.

## 4.2 Протокол испытаний

Структурные критерии используют модель программы в виде "черного ящика", что предполагает не знание исходного текста программы или спецификации программы в виде потокового графа управления.

### 4.2.1 Модуль «Добавления товара в чек»

**Таблица 6** -Структура данных функции «Добавления товара в чек»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void AddChek\_Click(object sender, RoutedEventArgs e) | Проверяет корректность введенных данных и добавляет товар в чек | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 22** – Общий алгоритм функции «Добавления товара в чек»

### 4.2.2 Модуль «Вывод чека»

**Таблица 7** -Структура данных функции «Вывод чека»

|  |  |  |
| --- | --- | --- |
| **Сигнатура** | **Описание** | **Параметры** |
| void Window\_Loaded(object sender, RoutedEventArgs e) | Загрузка данных на форму при открытии окна | Входные параметры являются служебными и в модуле практического применения не имеют |



**Рисунок 23** – Общий алгоритм функции «Вывод чека»

**Таблица 8** –Набор тестов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Описание | Шаги теста | Ожидаемые результаты | Реальные результаты | Прошел/  Провалился | Тестер/  Дата |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | Тестирование функции «Добавления товара в чек» | Не выбрать названия товара и не заполнить количество | Вывод ошибки о неправильном заполнении полей | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 2 | Тестирование функции «Добавления товара в чек» | Выбрать названия товара и не заполнить количество | Вывод ошибки о неправильном заполнении полей | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 3 | Тестирование функции «Добавления товара в чек» | Не выбрать названия товара и заполнить количество | Вывод ошибки о неправильном заполнении полей | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 4 | Тестирование функции «Добавления товара в чек» | Выбрать названия товара и заполнить количество меньше остатка | Добавление элемента в чек | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 5 | Тестирование функции «Добавления товара в чек» | Выбрать названия товара и заполнить количество больше остатка | Вывод ошибки о недостаточном количестве товара | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 6 | Тестирование функции «Добавления товара в чек» | Выбрать названия товара и заполнить количество отрицательное значение | Вывод ошибки о неправильном заполнении полей | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 7 | Тестирование функции «Показ чека» | Сума = 500 | Вывод цены со скидкой и без скидки | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |
| 8 | Тестирование функции «Добавления товара в чек» | Сума = 500 | Вывод цены только без скидки | Совпали с ожидаемыми | Прошел | Перевезенцева П.А. |

**ЗАКЛЮЧЕНИЕ**

В настоящее время широко в нашу жизнь внедряются информационные технологии. На сегодняшний день, наверное, не осталось никакой области, где бы они ни применялись. Одной из сфер деятельности, в которой информационные технологии получили широкое распространение, является сфера образования.

При выполнении задания по практике удалось достигнуть все поставленные цели. А именно были выполнены следующие этапы:

Обследование и разработка технического задания

* Построение блок-схемы программы
* Разработка программы.
* Тестирование и устранение ошибок.

Таким образом, мы приходим к следующему выводу - использование разработанной программы заметно поможет в образовательном процессе. Поможет новым людям познакомиться с мототехникой и приобщиться к ней.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. ГОСТ 19.201-78. Единая система программной документации. Техническое задание, требование к содержанию и оформлению [Текст]. - Введ. 2002-07-01. - М.: ФГУП «СТАНДАРТИНФОРМ», 2008. - 27с.
2. ГОСТ 19.505-79. Единая система программной документации. Руководство оператора. Требования к содержанию и оформлению [Текст]. - Введ. 2002-07-01. - М.: ФГУП «СТАНДАРТИНФОРМ», 2008. - 27с.
3. ГОСТ 19.301-79. Единая система программной документации. Программа и методика испытаний. Требования к содержанию и оформлению [Текст]. - Введ. 2002-07-01. - М.: ФГУП «СТАНДАРТИНФОРМ», 2008. - 27с.
4. ГОСТ 7.32-2001 СИБИД. Отчет о научно-исследовательской работе. Структура и правила оформления [Текст]. - Введ. 2002-07-01. - М.: ФГУП «СТАНДАРТИНФОРМ», 2008. - 27с.
5. ГОСТ 7.1-2003 СИБИД. Библиографическая запись. Библиографическое описание. Общие требования и правила составления [Текст]. - Введ. 2003-07-02. - М.: ФГУП «СТАНДАРТИНФОРМ», 2010. - 49с.

# ПРИЛОЖЕНИЕ А

public partial class MainWindow : Window

{

KafeEntities \_kafe = KafeEntities.GetContext();

List<string> \_listChek = new List<string> { " ЧЕК " };

int KolVoChek = 0;

int AllSumm = 0;

double Summ;

int SumObed = 0;

public MainWindow()

{

InitializeComponent();

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

\_kafe.Tovars.Load();

dataGrid.ItemsSource = \_kafe.Tovars.Local.ToBindingList();

NameAdd.ItemsSource = \_kafe.Tovars.ToList();

}

private void AddChek\_Click(object sender, RoutedEventArgs e)

{

var \_tovar = \_kafe.Tovars.Where( p => NameAdd.Text == p.Name).FirstOrDefault();

if (\_tovar.KolvoOst <= KolVoChek)

{

MessageBox.Show("Недостаточно товара");

}

else

{

if (\_tovar != null && KolVoChek >= 0)

{

if (\_tovar.Name == "Комплексный обед")

{

SumObed += Convert.ToInt32(KolVoChek \* \_tovar.PriceOne);

}

else

{

AllSumm += Convert.ToInt32(KolVoChek \* \_tovar.PriceOne);

if (300 <= AllSumm && AllSumm >= 500)

{

Summ = AllSumm - AllSumm / 100 \* 3;

}

else if (500 < AllSumm && AllSumm >= 1000)

{

Summ = AllSumm - AllSumm / 100 \* 5;

}

else if (1000 < AllSumm && AllSumm >= 5000)

{

Summ = AllSumm - AllSumm / 100 \* 7;

}

else if (5000 < AllSumm)

{

Summ = AllSumm - AllSumm / 10;

}

}

\_listChek.Add($"{\_tovar.Name} {\_tovar.PriceOne} Х {KolVoChek}{\_tovar.Units} {KolVoChek \* \_tovar.PriceOne}");

ShowChek.Visibility = Visibility;

}

else MessageBox.Show("Выберите название товара или его количество. Возможно данные были введены не корректно");

}

}

private void ShowChek\_Click(object sender, RoutedEventArgs e)

{

Data.Sum = Summ;

Data.AllSum = AllSumm;

Data.SumObed = SumObed;

Data.list = \_listChek;

Chek chek = new Chek();

chek.ShowDialog();

}

private void Returne\_Click(object sender, RoutedEventArgs e)

{

\_listChek.Clear();

\_listChek = new List<string> { " ЧЕК " };

AllSumm = 0;

Summ = 0;

SumObed = 0;

ShowChek.Visibility = Visibility.Hidden;

}

private void KolvoChek\_TextChanged(object sender, TextChangedEventArgs e)

{

if (int.TryParse(KolvoChek.Text, out int KolChek) ==true)

{

KolVoChek = KolChek;

} else KolVoChek = 0;

}

private void FindTovar\_Click(object sender, RoutedEventArgs e)

{

var findN = from p in \_kafe.Tovars

where p.Name.StartsWith(FindName.Text)

select new { p.Name, p.Units, p.KolvoOst, p.PriceOne };

dataGrid.ItemsSource = findN.ToList();

ReturneFind.Visibility = Visibility;

}

private void ReturneFind\_Click(object sender, RoutedEventArgs e)

{

dataGrid.ItemsSource = \_kafe.Tovars.Local.ToBindingList();

ReturneFind.Visibility = Visibility.Collapsed;

}

}

# ПРИЛОЖЕНИЕ В

public partial class Chek : Window

{

public Chek()

{

InitializeComponent();

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

if (Data.AllSum >= 300) { Summ.Visibility = Visibility; textPrice.Visibility = Visibility; }else { Summ.Visibility = Visibility.Collapsed; textPrice.Visibility = Visibility.Hidden; }

listBoxChek.ItemsSource = Data.list;

Summ.Text = (Data.Sum+Data.SumObed).ToString();

AllSumm.Text = (Data.AllSum+Data.SumObed).ToString();

}

private void Returne\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

}