**ОТЧЕТ**

**ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

ПМ.05 - Проектирование и разработка ИС

Специальность: 09.02.07 Информационные системы и программирование

|  |
| --- |
| **Студента гр.ИС-117**  Кугоян Саркис |
|  |
| **Организация**  ООО «Виктория» |
|  |
| **Руководитель практики**  Абметка Вероника Георгиевна |

*Сбор данных для анализа использования и функционирования информационной системы. Исследование организационной структуры*

Продуктовый магазин, в котором я проходил практику имел линейно-функциональную структуру управления  
 ***Преимущества линейно-функциональной структуры управления:***  
- Разумный баланс между соблюдением принципа единоначалия и рациональной специализацией отдельных уровней управленческой деятельности.

- Руководитель организации освобождается от необходимости подробного анализа проблем.

- Сохраняется возможность привлечения консультантов и экспертов для участия в подготовке и проработке управленческих решений.

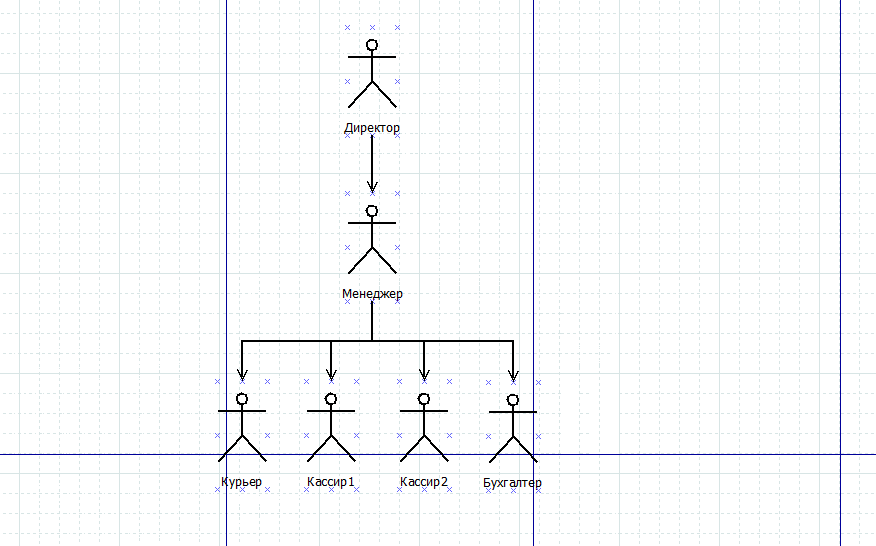
***Недостатки линейно-функциональной структуры управления:***

- Снижается степень взаимодействия между разными подразделениями организации на горизонтальных уровнях, координация осуществляется верхними управленческими звеньями.

- Снижение ответственности отдельных руководителей.

- Усиление вертикальной управленческой иерархии в ущерб качеству реализации управленческих решений.

На компьютере у них есть база данных «Доставка», служащая для хранения заказов



*Произведение оценки информационной системы для выявления возможности ее модернизации*

Магазин производил курьерскую доставку. Для хранения заказов есть база данных «Доставка», в которой хранятся заказы. Она состоит из следующих таблиц:

* Таблица «Заказы» для занесение данных о заказах. Она имеет следующие столбцы:
  + Наименование(nvarchar(50)
  + Единица измерения(nvarchar(50))
  + Кол-во(int)
  + Цена(money)
  + Сумма(money)
  + Дата доставки(nchar(10))
* Таблица «Постоянные клиенты» для хранения данных о постоянны покупателях. Она имеет следующие столбцы:
  + Фамилия(nvarchar(50))
  + Имя (nvarchar(50))
  + Отчество(nvarchar(250))
  + Номер телефона(nvarchar(12))
  + Скидка(int)

Модернизацией данной ИС (информационной системы) станет разработка приложения для удобной работы с БД (базой данных).

*Исследование предметной области*

Основную прибыль магазин получал от продажи хлебобулочных изделий и яиц.

В магазине работали: директор, менеджер, который контролирует большинство процессов. Так же работают 2 кассира, 1 курьер и 1 бухгалтер.

В данной работе будет разрабатываться приложение для работы с БД «Заказы».

*Сбор и анализ данных для разработки проектной документации на ИС*

Вместе с менеджером и курьером предприятия мы проанализировали работу курьера и составили ТЗ (техническое задание) на разработку приложения.

Доставка происходит следующим образом:

1. Покупатель делает заказ по телефону
2. Курьер записывает всю информацию в базу данных и назначает дату и время доставки
3. По информации в базе данных производиться доставка

*Разработка проектной документации на оптимизацию ИС предприятия*

Техническое задание на разработку приложения для работы с БД

1. *Терминология.*
   1. БД – база данных
   2. ИС – информационная система
2. *Цель.*

Целью для создания программы является более удобная работа c ИС.

1. *Требования к программе.*

Программа должна иметь следующие функции:

1. Авторизация;
2. Показывать данные о заказах и иметь возможность их поиска, редактирования и удаления;
3. Иметь отдельное окно для просмотра постоянных клиентов и иметь возможность их поиска, редактирования и удаления;
4. Обработка исключительных ситуаций

Также программа должна иметь понятный интерфейс.

1. *Сценарии использования*

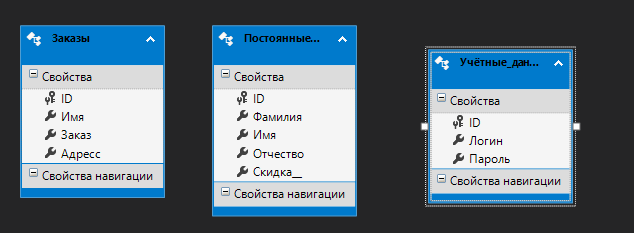
Логин и пароль хранятся в базе данных. При входе в приложения, должна открываться форма авторизации, на которой необходимо ввести логин в поле «Логин» и пароль в поле «Пароль». Далее, открывается главная форма, в которой должна быть информация о заказах. С неё можно переходить на вторую форму, в которой должна быть информация о постоянных покупателях, и возвращаться обратно.

1. *Описание экранов*

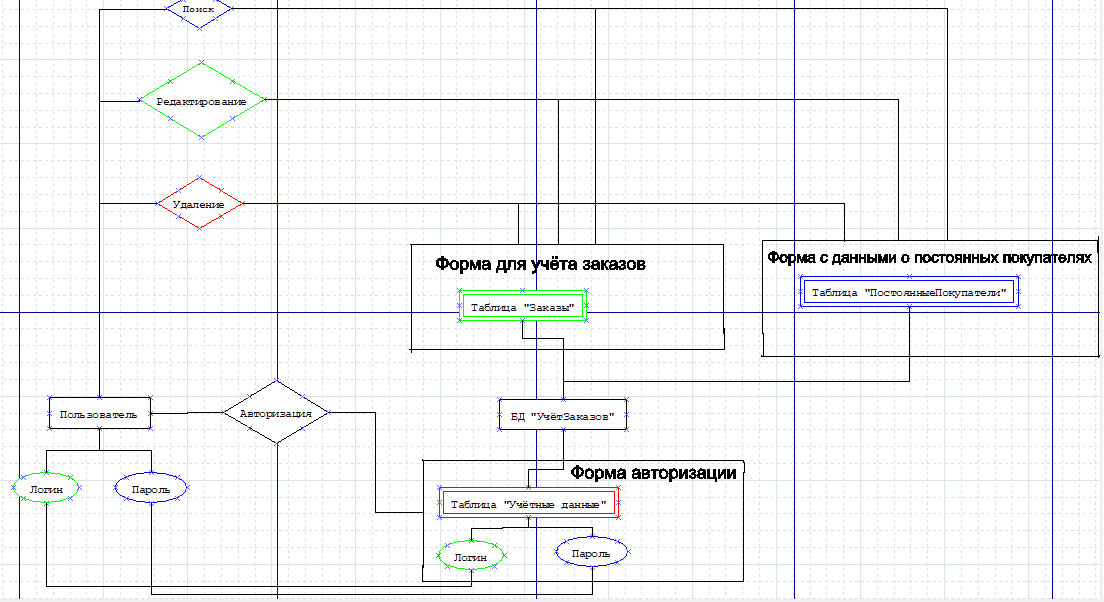
На главной форме выводится информация о заказах и есть возможность поиска, редактирования записей и их удаление. Так же на форме должна быть кнопка для перехода на вторую форму, где будет список постоянных покупателей, должна быть возможность поиска, редактирования и удаления записей.

*Моделирование ИС*

*Рис 1. Модель БД*



*Рис 2. Модель ИС*



На данной модели показано как должна работать программа. При запуске приложения открывается форма для авторизации, где мы вводим логин и пароль. Верные логин и пароль хранятся в БД «УчётЗаказов» в таблице «Учётные данные». При авторизации проверяется соответствие введённых данных с теми, которые находятся в БД. При верно введённом логине и пароле, открывается главная форма для учёта заказов, с которой возможен переход на вторую форму с данными о постоянных клиентах. При работе с данными формами можно выполнять поиск, редактирование и удаление данных в БД.

*Выбор среды разработки программных средств Расчет экономической эффективности*

Для разработки программного средства была выбрана среда разработки Visual Studio и язык программирования C#.

Для расчёта экономической эффективности была выбрана популярная методика «Возврат инвестиций» (Return of Investment, ROI). Затрачивая на ИТ-проекты немалые средства, руководители предприятия хотят знать, когда получат их обратно. Анализ этого показателя рассматривается как способ продемонстрировать необходимость вложения средств в информационные технологии. ROI показывает отношение увеличения инвестиций (чистой прибыли) к объему инвестиций:

ROI = (Доход / Расход) \* 100%

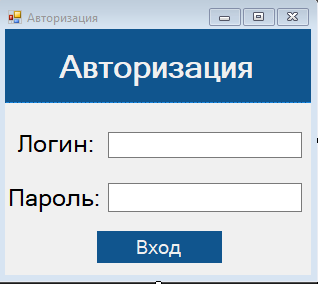
На данный момент доставка продуктов приносит около 5000р. в месяц. После внедрения программного средства, доход может увеличиться до 5% (250р.), так как курьеру будет легче выполнять работу и он сможет развозить больше продуктов. Расходов на создание программного средства не было. Таким образом, мы можем посчитать ROI:

ROI = 250 \* 100% = 250%

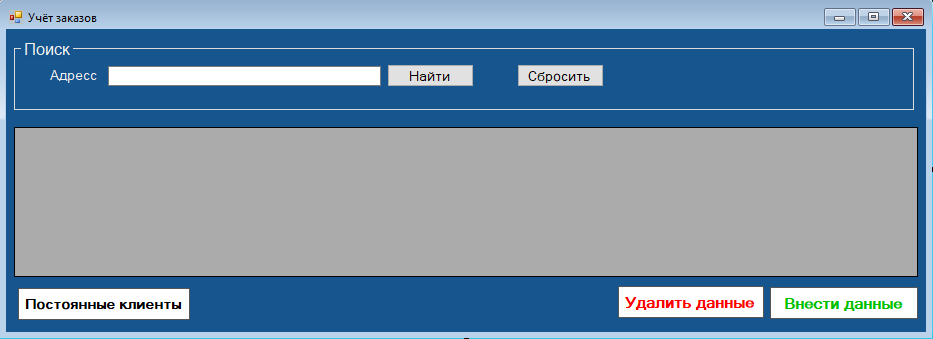
*Разработка пользовательского интерфейса*

Пользовательский интерфейс был разработан с помощью конструктора Windows Forms. Он упрощает работу над созданием пользовательского интерфейса.

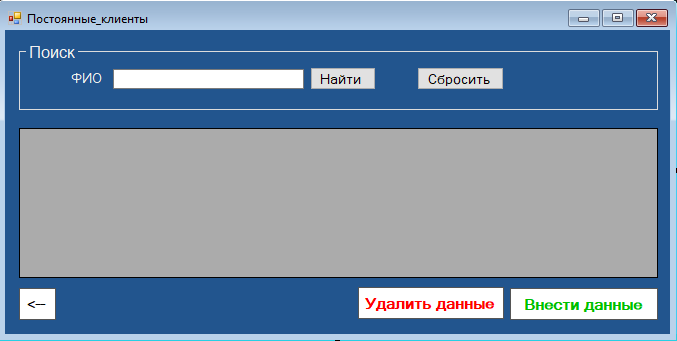
*Рис 3. Окно авторизации*



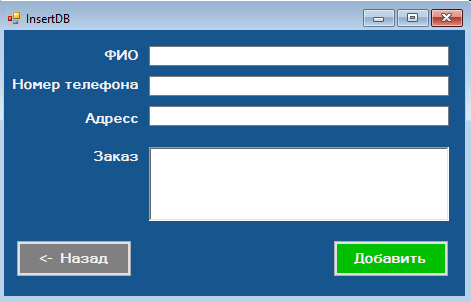
*Рис 4. Окно работы с заказами*



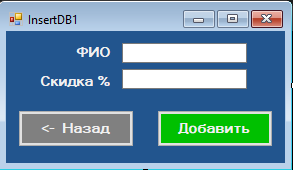
*Рис 5. Окно работы постоянными клиентами*



*Рис 6. Окно занесения данных в БД в таблицу «Заказы»*



*Рис 7. Окно занесения данных в БД в таблицу «ПостоянныеПокупатели»*



*Разработка модулей ИС в соответствие с техническим заданием*

Был выбран метод нисходящей разработки. Нисходящий подход предполагает, что проектирование и последующая реализация компонентов выполняется «сверху-вниз», т.е. вначале проектируют компоненты верхних уровней иерархии, затем следующих и так далее до самых нижних уровней.

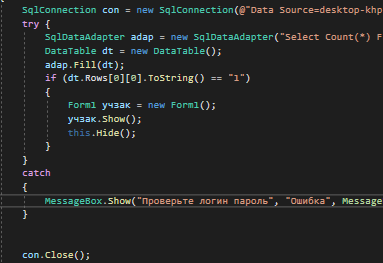
Модули ИС разрабатывались в следующем порядке:

1. «Авторизация»
2. «Учёт заказов»
3. «Постоянные\_клиенты»
4. InsertDB. Отвечает за внесение данных в БД в таблицу «Учёт заказов»
5. InsertDB1. Отвечает за внесение данных в БД в таблицу «ПостоянныеПокупатели»

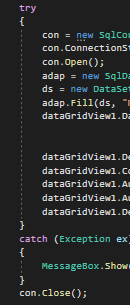
*Разработка подсистемы безопасности ИС в соответствии с техническим заданием*

В соответствии с ТЗ (техническим заданием) у нас есть окно авторизации. Так же были обработаны исключительные ситуации с помощью try{} catch{}.

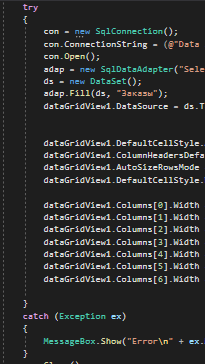
*Рис 8. Обработка исключительных ситуаций модуля «Авторизация»*



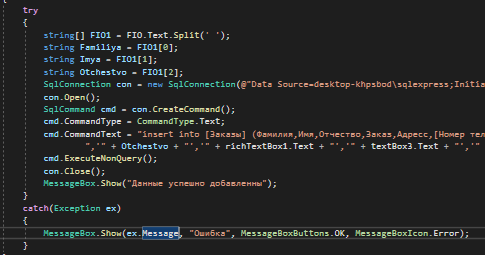
*Рис 9. Обработка исключительных ситуаций модуля «Учёт заказов»*



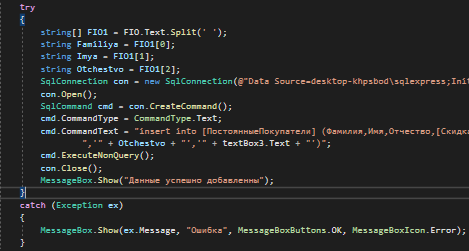
*Рис 10. Обработка исключительных ситуаций модуля «ПостоянныеПокупатели»*



*Рис 11. Обработка исключительных ситуаций модуля «InsertDB»*



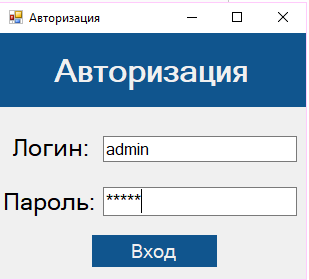
*Рис 12. Обработка исключительных ситуаций модуля «InsertDB1»*



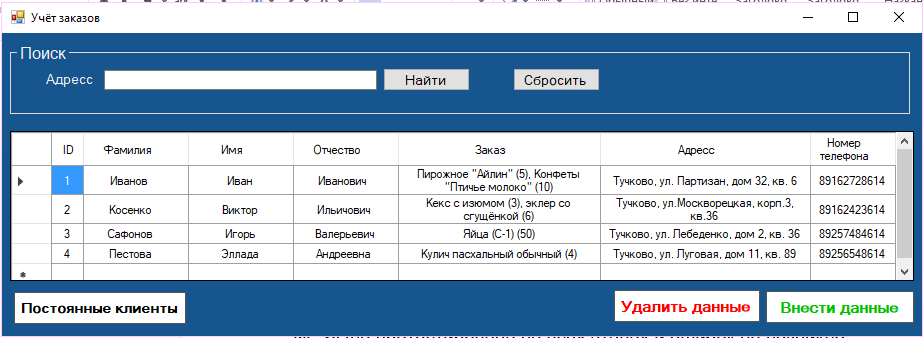
*Осуществление тестирования ИС на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях ИС. Тестирование ИС на этапе внедрения. Тестирование ИС на этапе сопровождения*

ИС была протестирована на всех этапах и ошибок не возникло.

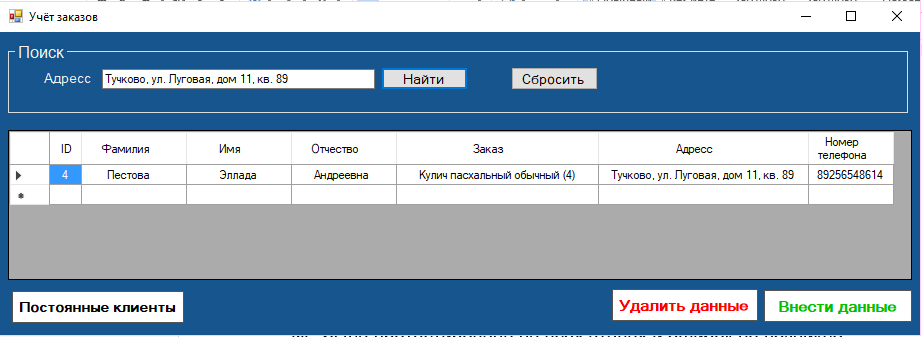
*Рис 13. Процесс авторизации*



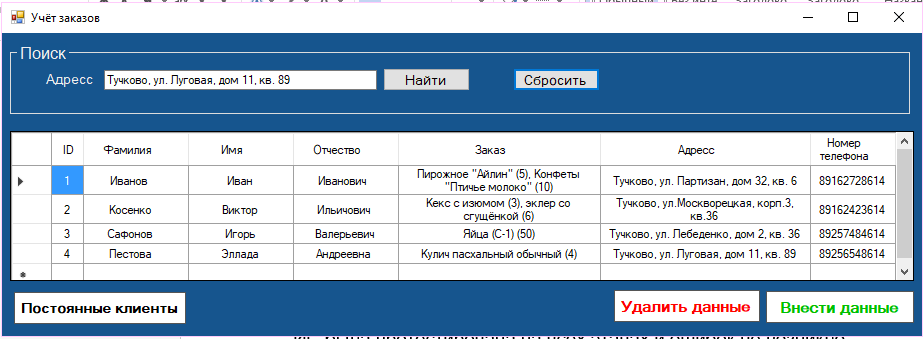
*Рис 14. Окно «Учёт заказов»*



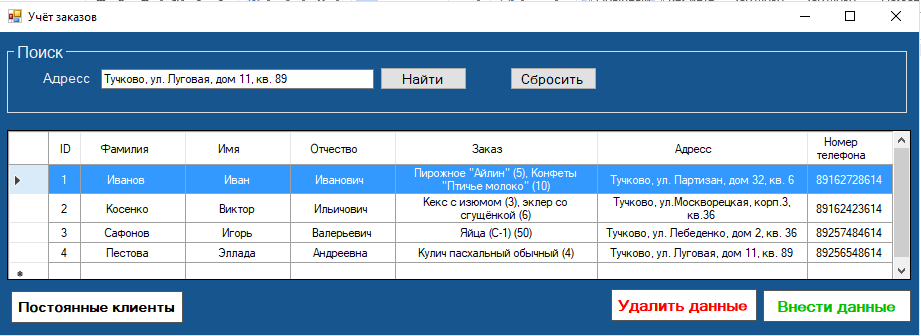
*Рис 15. Поиск в окне «Учёт заказов»*



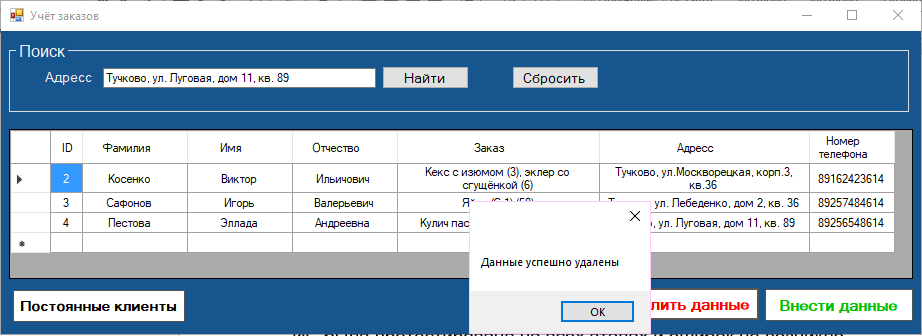
*Рис 16. Сброс поиска в окне «Учёт заказов»*



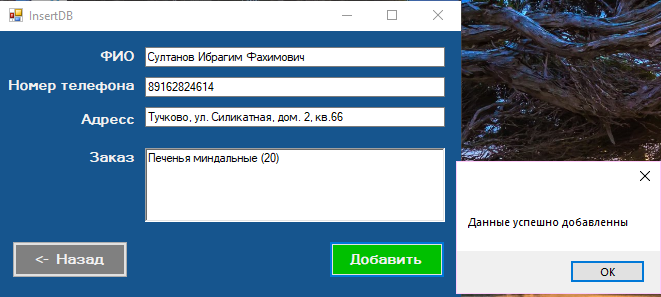
*Рис 17. Удаление данных в окне «Учёт заказов». Выбираем строчку*



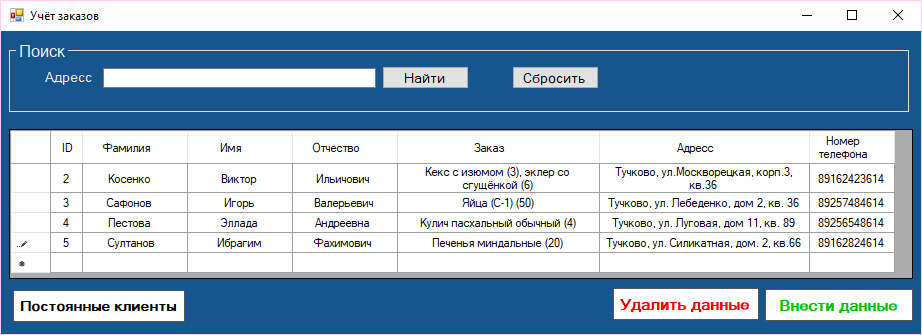
*Рис 18. Удаление данных в окне «Учёт заказов». Нажимаем на кнопку «Удалить данные». Нажимаем «ОК»*



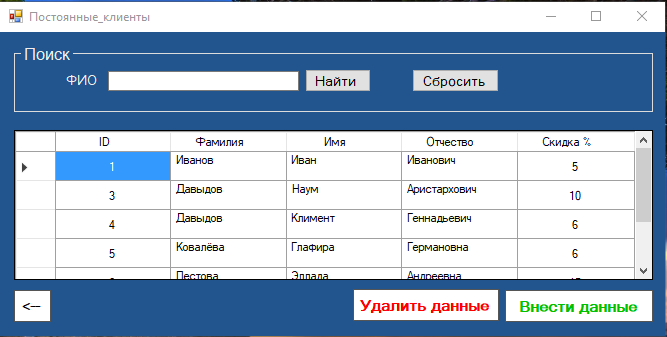
*Рис 19. Занесение данных через окно «Учёт заказов». Нажимаем на кнопку «Внести данные», заполняем нужные поля и нажимаем «Добавить». Нажимаем «ОК». Нажимаем «Назад»*



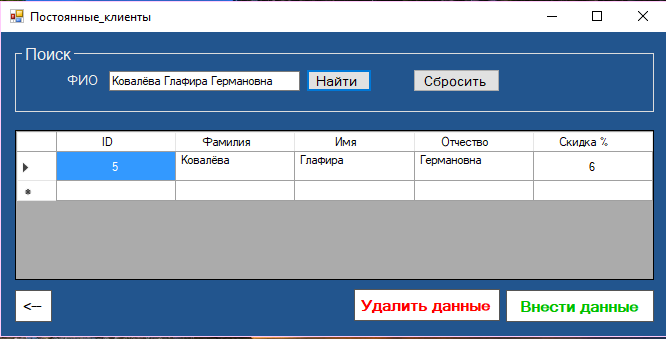
*Рис 20. Видим, что запись добавилась.*



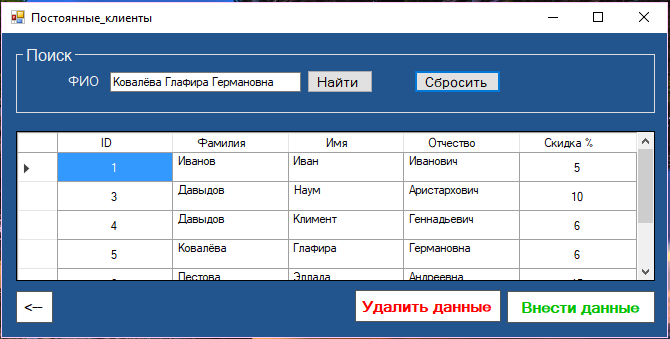
*Рис 21. На окне «Учёт заказов» нажимаем на кнопку «Постоянные клиенты» и выполняется переход на окно «Постоянные\_клиенты».*



*Рис 22. Поиск в окне «Постоянные\_клиенты».*



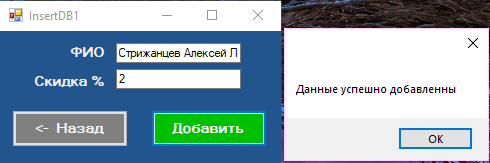
*Рис 23. Сброс поиска в окне «Постоянные\_клиенты».*



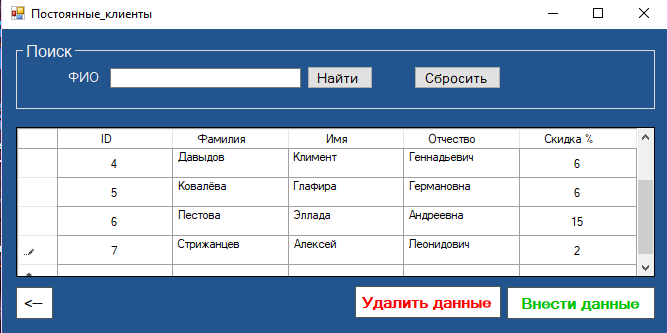
*Рис 24. Удаление данных в окне «Постоянные\_клиенты». Происходит аналогично удалению на окне «Учёт заказов»*



*Рис 25. Занесение данных в окне «Постоянные\_клиенты». Происходит аналогично занесению данных на окне «Учёт заказов»*



*Рис 26. Нажимаем на кнопку «Назад» на окне «InsertDB1» и видим, что данные добавлены*



*Разработка документации на эксплуатацию ИС*

Для ИС было составлено руководство пользователя. Оно приложено ниже.

