

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(национальный исследовательский университет)  
Высшая школа электроники и компьютерных наук  
Кафедра «Электронные вычислительные машины»

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой, доцент, к. т. н.

\_\_\_\_\_ К.А. Домбровский

« \_\_\_\_ » \_\_\_\_\_ 2017 г

Разработка автоматизированной информационной системы автомобильной  
газозаправочной станции

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА  
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ  
ЮУрГУ- 09.03.01.2017.265 ПЗ ВКР

Руководитель проекта, доцент, к. т. н.

\_\_\_\_\_ К. А. Домбровский

« \_\_\_\_ » \_\_\_\_\_ 2017 г.

Автор проекта

студент группы КЭ-445

\_\_\_\_\_ В. А. Руди

« \_\_\_\_ » \_\_\_\_\_ 2017 г.

Нормоконтролер, ст. преп. каф.

«Электронные вычислительные  
машины»

\_\_\_\_\_ В. В. Лурье

« \_\_\_\_ » \_\_\_\_\_ 2017 г.

Челябинск 2017

## Аннотация

Руди В.А. Разработка автоматизированной информационной системы автомобильной газозаправочной станции. – Челябинск: ФГБОУ ВПО «ЮУрГУ» (НИУ) ВШЭКН; 2017, 76 с., 29 ил., 3 табл., библиогр. список – 23 наим., 4 прил.

Работа посвящена разработке и проектированию информационной системы, автоматизирующей деятельность автомобильной газозаправочной станции.

Данная работа состоит из введения, трех глав, заключения, библиографического списка, четырех приложений.

В первой главе представлен обзор общих понятий и определений, связанных с построением информационной системы организации. Вторая глава посвящена описанию исследуемого предприятия, анализу его основных бизнес процессов и обоснованию потребности в автоматизации. Третья глава посвящена поэтапной разработке информационной системы, а также ее тестированию.

В заключении приведено описание основных результатов работы.

Приложения содержат скрипт инициализации базы данных, макеты форм пользовательского интерфейса, диаграмму классов, а также листинги программных модулей.

					<b>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</b>			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>				
<i>Разраб.</i>	В.А. Руди				<i>Разработка автоматизированной информационной системы автомобильной газозаправочной станции</i>	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Пров.</i>	К.А. Домбровский					<i>Д</i>	3	76
<i>Н. контр.</i>	В.В. Лурье					ФГБОУ ВПО «ЮУрГУ» (НИУ) Кафедра ЭВМ		
<i>Утв.</i>	К.А. Домбровский							

## Оглавление

ВВЕДЕНИЕ .....	6
1 ОБЩИЕ ПОНЯТИЯ И ПРИНЦИПЫ ПОСТРОЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ОРГАНИЗАЦИИ .....	8
1.1 Понятие и виды информационных систем.....	8
1.2 Общие принципы построения ИС .....	11
1.3 Особенности внедрения ИС на предприятии.....	16
2 ОПИСАНИЕ «АГЗС+» И ЕГО ОСНОВНЫХ БИЗНЕС ПРОЦЕССОВ .....	18
2.1 Общее описание предприятия .....	18
2.2 Анализ организационной структуры .....	19
2.3 Анализ основных бизнес-процессов компании .....	21
2.3.1 Построение модели AS-IS в нотации IDEF0.....	22
2.3.2 Функциональное моделирование в нотации DFD.....	23
2.4 Обоснование необходимости разработки ИС .....	25
3 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ.....	29
3.1 Постановка задачи .....	29
3.1.1 Основание для разработки .....	29
3.1.2 Назначение.....	29
3.2 Требования к программе или программному изделию.....	29
3.2.1 Требования к функциональным характеристикам .....	29
3.2.2 Требования к видам обеспечения.....	30
3.2.3 Требования к надежности .....	31
3.2.4 Требования к составу и параметрам технических средств.....	31
3.2.5 Требования к информационной и программной совместимости .....	31
3.2.6 Требования к программной документации.....	31
3.3 Анализ существующего программного обеспечения на рынке .....	31
3.4 Выбор и обоснование инструментальных средств для решения поставленной задачи.....	35
3.4.1 Выбор СУБД.....	35
3.4.2 Выбор языка программирования и среды разработки .....	36
3.5 Разработка инфологической и даталогической моделей данных .....	38

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

3.5.1 Инфологическая модель.....	38
3.5.2 Даталогическая модель базы данных .....	40
3.5.3 Физическая модель данных .....	41
3.6 Разработка интерфейса информационной системы .....	42
3.7 Разработка программных модулей.....	45
3.8 Тестирование разработанных компонентов информационной системы ...	50
ЗАКЛЮЧЕНИЕ.....	60
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	62
ПРИЛОЖЕНИЕ А – СКРИПТ ИНИЦИАЛИЗАЦИИ БД.....	64
ПРИЛОЖЕНИЕ Б – МАКЕТЫ ФОРМ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА .....	70
ПРИЛОЖЕНИЕ В – ЛИСТИНГИ ПРОГРАММНЫХ МОДУЛЕЙ.....	71
Листинг 1 – Функции работы с БД.....	71
Листинг 2 – Класс выгрузки данных в Excel.....	74
ПРИЛОЖЕНИЕ Г – ДИАГРАММА КЛАССОВ .....	76

## ВВЕДЕНИЕ

На сегодняшний день в мире информационные технологии достигли высокого уровня развития. Внедрение новейших технологий помогает организовать рабочий процесс эффективнее, быстрее, надежнее. Рост конкуренции принуждает каждую компанию, организацию использовать новейшие подходы в оказании услуг. Рост операций, расширение услуг, увеличение объема торговли вынуждает компании совершенствовать управленческую деятельность.

В связи с вышесказанным большинство компаний используют в своей деятельности автоматизированные системы, что позволяет эффективно хранить, обрабатывать и использовать накопленную информацию. Отметим, что эффективная работа компании очень зависит от уровня оснащения компании информационными системами.

Работа актуальна тем, что на предприятии «АГЗС+» основной объем документооборота, касающийся деятельности предприятия выполняется при помощи табличного процессора Excel, что очень неэффективно и затрудняет оперативный учет и анализ данных.

Целью данной работы является проектирование информационной системы, автоматизирующей деятельность АГЗС.

Для достижения поставленной цели необходимо разработать автоматизированную информационную систему (АИС) управления, которая должна обеспечивать:

- организацию внутреннего информационного обмена в рамках процесса управления предприятием;
- необходимый уровень безопасности и защиты информационных ресурсов предприятия.

Объектом исследования является предприятие «АГЗС+».

Предметом исследования являются способы повышения эффективности деятельности предприятия посредством внедрения АИС.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

Для достижения поставленной цели решены следующие задачи:

- рассмотрены общие понятия и принципы построения информационной системы организации;
- проанализирована деятельность предприятия;
- выявлены недостатки существующего подхода;
- произведено проектирование и тестирование АИС.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дата		

# 1 ОБЩИЕ ПОНЯТИЯ И ПРИНЦИПЫ ПОСТРОЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ ОРГАНИЗАЦИИ

## 1.1 Понятие и виды информационных систем

В качестве официального определения информационной системы (ИС) можно рассматривать определение, которое дает Федеральный закон Российской Федерации от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации»:

«Информационная система – совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств».

Таким образом, информационной системой называют такую систему обработки информации и такие информационные ресурсы, которые способны обеспечить и распространить информацию.

К таким организационным ресурсам относятся человеческие, технические, финансовые и т. д. ресурсы.

Таким образом, к информационным системам относится такая совокупность средств, методов и персонала, которая взаимосвязана между собой и применяется не только для хранения информации, но и для ее обработки и выдачи, достигая при этом определенной цели управления. Ни для кого не будет большим секретом, что в современном мире основным средством обработки этой информации, конечно же, является компьютер. Так же следовало бы отметить, что в основном информационные системы преобразуют не столько информацию, сколько данные и поэтому правильнее будет, если мы будем называть эти системы не информационными, а системами обработки данных.

### Основные функции и процессы в информационной системе

Теперь рассмотрим более подробно функции, используемые в информационных системах:

- во-первых, основной функцией информационных систем, конечно же, является сбор информации;

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						8
Изм.	Лист	№ докум.	Подпись	Дата		

- во-вторых, немаловажной функцией является функция хранения этой информации;
- в-третьих, в информационных системах никак не обойтись без поиска и сбора различной информации;
- в-четвертых, немаловажную роль играет и передача информации.

То есть если говорить кратко, то в информационной системе происходят следующие процессы:

- ввод информации от источников информации;
- обработка (преобразование) информации;
- хранение входящей и обработанной информации;
- вывод информации для правок пользователю;
- ввод информации от пользователя через обратную связь.

#### Определение структуры информационной системы

Основу системного построения информационной системы составляет ее структура, которая должна быть достаточно полной. Средства структуризации есть процедуры декомпозиции (анализа) и композиции (синтеза) системы. Поэтому важным этапом проектирования информационной системы является структуризация системы – локализация ее границ и выделения структурных составных частей.

В современной информационной системе компьютер играет роль аппаратно-программной части.

Аппаратная часть: (Hardware, «твердая часть» – «железо») состоит из соединенных между собой различных устройств, таких как: процессор, запоминающее устройство, внешние устройства ввода-вывода, контроллеры внешних устройств.

Программное обеспечение (ПО): (Software, «мягкая») состоит из набора различных программ, управляющих работой компьютера, поддерживают диалог с пользователем, обрабатывают разнообразную информацию, помогают



создавать различные программы.

Аппаратная часть не может выполнять операции по обработке информации без различных программ, благодаря которым устройства выполняют свои функции. Для обеспечения работоспособности компьютера и выполнения определенной работы нужна совокупность программ, которая создает программное обеспечение.

Программное обеспечение осуществляет управление устройствами компьютера во время ввода, обработки, вывода и хранения информации, создает условия для работы пользователя на компьютере и тому подобное.

Автоматизированная информационная система

Автоматизированная ИС – система, реализующая информационную технологию в сфере управления для совместной работы комплекса технических средств с управленческим персоналом. Ее предназначение – автоматизированный сбор, регистрация, хранение, поиск, обработка и выдача информации по запросам пользователей (управленческого персонала).

ИС должна обеспечивать:

- постоянное наблюдение за текущим состоянием объекта управления и его характеристиками;
- адаптацию, то есть приспособление к принятой практике бизнеса, и модификацию, если такая практика меняется;
- общую поддержку профессиональной деятельности управленческих работников;
- взаимодействие с руководством;
- осуществление сбора и анализа данных для управления и автоматического выполнения программными средствами при наступлении заданного времени с формированием необходимой отчетности;
- реализацию удобной системы подсказок и рекомендаций для пользователей;

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		10

- эффективное сохранение данных в базе данных и возможность доступа к ним любого пользователя со своего рабочего места;
- взаимодействие пользователей между собой на основе непрерывной технологии.

В зависимости от целей выделяют такие типичные функции ИС как: прогнозирование, планирование, учет, контроль, анализ, регулирование.

### Классификация автоматизированных ИС

Автоматизированные ИС можно классифицировать по типу поддержки, которую они обеспечивают организации.

Существуют системы первого класса или системы обеспечения операций, которые обрабатывают генерируемую информацию и применяется в деловых операциях, и системы второго класса или системы обеспечения менеджмента, которые оказывают помощь менеджерам в принятии решений.

Системы 1-го класса делятся на 3 группы: системы обработки операций, автоматизированные системы управления технологическими процессами, системы сотрудничества на предприятии.

Системы 2-го класса делятся на такие виды: информационные менеджерские системы – системы обеспечения менеджмента, системы поддержки принятия решений, управленческие.

### 1.2 Общие принципы построения ИС

Построение информационных систем представляет собой процесс создания и внедрения проектов по комплексному решению экономических задач, используя новые технологии.

В этот процесс включается детальная разработка проектных решений, а также их анализ, апробация, внедрение. Качественное проектирование и внедрение, несомненно, являются основной предпосылкой для эффективного функционирования системы при условии постоянного совершенствования ее функциональных и обеспечивающих составляющих.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						11
Изм.	Лист	№ докум.	Подпись	Дата		

На рисунке 1.1 представлена структура процесса построения ИС.



Рисунок 1.1 – Проектирование информационных систем

Целью всех этих работ служит не только компьютеризация информационных потоков, но и совершенствование самого управления, а также организация основной деятельности рассматриваемого экономического объекта. Следовательно, необходимо иметь представление об основных имеющихся на рынке технических и программных средствах, а также тенденциях их развития, основных принципах построения.

Поскольку практически все современные ИС являются автоматизированными ИС, то и основные принципы проектирования ИС будут рассматриваться с точки зрения создания ИС.

Перечислим далее основные принципы внедрения и функционирования ИС, которые следует учитывать при построении ИС.

Принцип идентичности. Разработка новой, совершенствование существующей или внедрение полученной (приобретенной) извне ИС являются идентичными научно-техническими проблемами, отличающимися друг от друга

содержанием ряда этапов и временными параметрами.

**Принцип технологичности.** Автоматизированная технология означает создание (разработку) новой технологии или модернизацию существующей (автоматизированной). При этом не должно быть простого использования автоматизированной ИС в условиях старых традиционных технологий.

**Непрерывность, поэтапность, преемственность** разработки и развития. ИС в общем случае представляет собой постоянно развивающуюся систему. Каждое нововведение должно служить развитию основных системных принципов и улучшению достигнутых параметров.

**Адаптивность.** Компоненты ИС должны обладать свойствами, обеспечивающими быстрое их приспособление к изменениям внешней среды, новым средствам и т.п.

**Модульный принцип** построения программных и технических средств предполагает, что указанные средства состоят из блоков (модулей), обеспечивающих возможность их замены или изменения с целью совершенствования ИС или её адаптации к новым условиям.

**Технологическая интеграция** предполагает для всей системы применение единой технологии создания, обновления, сохранения и использования информационных ресурсов. Например, однократную обработку информационных документов и их многократное, многоцелевое использование.

**Полная нормализация процессов и их мониторинг.** Многоцелевое использование информации ИС требует обеспечения высокой достоверности данных в системе. Для этого на различных этапах обработки и ввода информации необходимо использовать разные формы её контроля, требования к которому можно сформировать исходя из состава решаемых задач и обрабатываемых данных. Постоянный мониторинг необходим для получения качественных и количественных характеристик функционирования ИС на основе применения встроенных или используемых в виде отдельного модуля средств статистики.

**Регламентация.** ИС ориентированы на функционирование в промышленном

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		13

режиме, обеспечивающем массовую поточную обработку информационных документов. Эта обработка регламентируется стандартами, маршрутными и пооперационными технологиями, нормативами на ресурсные и временные показатели, развитой службой диспетчеризации.

Экономическая целесообразность. Создание ИС должно предусматривать использование проектных решений, обеспечивающих минимизацию финансовых, материальных затрат и трудовых ресурсов, а также способствующих совершенствованию обслуживания пользователей.

Типизация или максимальное использование готовых решений и средств необходимо для сокращения стоимости, сроков разработки и внедрения ИС, а также уменьшения ошибок проектирования как системы в целом, так и отдельных её составляющих.

Стандартизация проектных решений предполагает, что разработку, развитие ИС и их сетей следует осуществлять с ориентацией на сотрудничество и кооперацию, а также в соответствии с правилами и протоколами национальных и международных стандартов.

Принцип корпоративности. При проектировании автоматизированной системы следует предусмотреть её аппаратную, программную, лингвистическую и информационную совместимость с другими ИС. Как правило, разные организации входят в состав различных систем и сетей (республики, края, области, города, района, ведомства и т.п.), участником которых они являются или могут стать. Требования корпоративности могут входить в противоречие с требованиями или решениями, диктуемыми другими принципами, например – преемственности проектных решений.

Ориентация на первых лиц объекта автоматизации. Успешное выполнение работ по созданию ИС, её развитию и эксплуатации возможно при условии их безусловной поддержки первым лицом (директор организации) и закреплении непосредственной ответственности за их выполнение приказом по организации за руководителем на уровне не менее заместителя директора. В подразделениях организации ответственность за выполнение работ должна возлагаться на

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						14
Изм.	Лист	№ докум.	Подпись	Дата		

руководителей этих подразделений.

Рассмотрим некоторые из названных принципов.

Преемственность – принцип проектирования новых функциональных возможностей системы. Он заключается в обязательном учёте в новых проектных решениях ранее накопленного опыта, а также сохранения всех полезных для дальнейшего использования ресурсов и средств. В первую очередь это относится к информационным ресурсам, ранее действовавшим версиям ИС, средствам лингвистического обеспечения, а также имеющимся в наличии техническим средствам.

Непрерывность и поэтапность развития. Это не только принцип проектирования, но и одно из наиболее важных свойств ИС, которая не может длительное время не видоизменяться.

Как показывает практика, развитие ИС сложнее, чем её проектирование и сопровождение. Стыковка ИС с уже действующими в режиме промышленной эксплуатации системами может оказаться сложной задачей. Чтобы не снижался уровень и качество функционирования системы необходимо проведение опытной эксплуатации.

Преемственность развития предполагает, наряду с сохранением ранее наработанного опыта и ресурсов, поэтапное развитие ИС. Сказанное относится к функциям системы, созданию и (или) внедрению новых средств информационного, программно-технологического, словарно-терминологического и технического обеспечения, а также сохранению ставших привычными для персонала ИС, машинных операций, визуальных форм рабочих листов и экранов.

Адаптивность рассматривается как заложенная в проектные решения возможность перестройки системы или отдельных ее составляющих «на ходу», т.е. без остановки эксплуатации ИС. Это одно из наиболее значимых свойств информационных систем.

Реализация модульного характера системы имеет свои преимущества и недостатки. Модульная структура программного обеспечения позволяет вести

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						15
Изм.	Лист	№ докум.	Подпись	Дата		

поэтапное внедрение пакетов прикладных программ, что развивает возможности системы, ускоряет ее внедрение, а также гарантирует автономное применение каждого пакета в других ИС. К отрицательным моментам относятся избыточность и раздробленность пакетов программ. Это усложняет эксплуатацию, требует наличия промежуточных носителей, дополнительных архивов, обеспечивающих надежность хранения массивов и увеличивает трудоемкость эксплуатации.

### 1.3 Особенности внедрения ИС на предприятии

Внедрение корпоративной ИС, разработанной самостоятельно или приобретенной у поставщика, зачастую сопровождается ломкой (перепроектированием) существующих на предприятии бизнес-процессов.

Нужно четко понимать, что корпоративная ИС призвана упростить управление организацией, улучшить процессы, усилить контроль и обеспечить этим конкурентные выгоды. Только с такой точки зрения можно оценивать пользу от ее внедрения.

К настоящему времени сложился стандартный набор приемов внедрения ИС. Основное правило: выполнять обязательные фазы последовательно и не пропускать ни одной из них.

Критически важными для внедрения являются следующие факторы:

- наличие четко сформулированных целей проекта и требований к ИС;
- наличие стратегии внедрения и использования ИС;
- проведение предпроектного обследования предприятия и построения моделей «Как есть» и «Как будет»;
- планирование работ, ресурсов и контроль выполнения плана внедрения;
- участие высшего руководства во внедрении системы;
- проведение работ по внедрению ИС специалистами по интегрированию систем совместно со специалистами предприятия;

- регулярный мониторинг качества выполняемых работ;
- быстрое получение положительных результатов хотя бы в части внедренных модулей ИС или в процессе ее опытной эксплуатации.

Этапы внедрения ИС на предприятии описываются в ГОСТ 34.601-90 «Автоматизированные системы. Стадии создания».

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		17



## 2 ОПИСАНИЕ «АГЗС+» И ЕГО ОСНОВНЫХ БИЗНЕС ПРОЦЕССОВ

### 2.1 Общее описание предприятия

За последние несколько лет во всем мире резко вырос интерес к газовому топливу. Объясняется это увеличением цен на нефть, экологическими требованиями, желанием правительств экономически развитых стран перейти на возобновляемые виды энергии.

Эффективность газового топлива доказана многочисленными исследованиями. Основными преимуществами газа являются:

- экономичность;
- увеличение межремонтного периода работы двигателя;
- увеличение срока службы моторного масла;
- увеличение срока службы свечей зажигания;
- снижение уровня шума работы двигателя;
- сокращение токсичности выхлопных газов.

Предприятие «АГЗС+» уже более 2-х лет работает на рынке услуг по заправке автомашин качественным газовым топливом.

Особое внимание компания уделяет качественному сервису и удобству клиентов. На заправочной станции предприятия можно в кратчайшие сроки заправить баллон или приобрести высококачественный газ.

Благодаря наличию собственной газонаполнительной станции и лаборатории компания может всегда отвечать за качество поставляемого товара. К тому же контроль качества относится и к такому параметру, как точный налив непосредственно на АГЗС.

Среди прочего оборудования компания имеет такие агрегаты, как:

- газонаполнительные станции;
- дополнительные фильтры;
- оборудование для проверки газа.

Все это позволяет получить действительно качественный метан, который по

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		18

достоинству оценят клиенты АГЗС. Регулярные проверки и калибровка оборудования позволяют убедиться в точности данных относительно его работоспособности.

Компания имеет собственную техническую службу, которая проводит регулярные проверки качества установленных агрегатов, а также поставку топлива клиентам. Все это позволяет избежать утечек и уменьшает издержки производства, что положительно сказывается на стоимости реализуемого газа. К тому же данные мероприятия позволяют заблаговременно выявить возможные сбои в подаче топлива или поломки, что не допускает экстренной остановки работы станции. АГЗС компании готовы принять клиентов в любое время суток и предоставить действительно качественное топливо.

Каждый автомобиль, заправляющийся на АГЗС, может быть уверен в том, что качество топлива соответствует всем техническим нормам и стандартам. Оно позволит не только сэкономить бюджет, но и сохранить все составные части двигателя в хорошем состоянии на протяжении долгого времени. Все это возможно благодаря регулярным проверкам поставляемого и реализуемого топлива.

По результатам потребления топлива клиентам выдается бонусная карта со скидкой. Автоматически рассчитывается сумма за заправку на основе потребленного топлива (литров), его вида и скидки данному клиенту.

## 2.2 Анализ организационной структуры

Организационная структура предприятия представлена на рисунке 2.1.

Организационная структура является линейной.

Произведем анализ организационной структуры «АГЗС+» по следующим характеристикам:

- оптимальное количество подчиненных сотрудников;
- однородность организационной структуры;
- перенесение должностных обязанностей;
- избыточность руководящего персонала.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		19



Рисунок 2.1 – Организационная структура предприятия

#### Оптимальное количество подчиненных сотрудников

В рассматриваемой оргструктуре директору административно подчинено 4 человека. Менеджеру также подчинены 4 человека. Согласно требованию оптимальности, число непосредственно подчиненных сотрудника должно быть не более 9. Требование выполняется.

#### Однородность организационной структуры

Рассматриваемая оргструктура «АГЗС+» является однородной.

#### Перенесение должностных обязанностей

Рассматриваемая оргструктура не имеет должностей, которые явно дублируют друг друга.

#### Избыточность руководящего персонала

Численность руководящего персонала составляет 3 человека. В среднем численность работников составляет 9 человек. Поэтому доля руководящего персонала составляет  $3/9=30\%$ , что равно пороговому значению в 30%, выше которого численность руководящего персонала считается избыточной.

### 2.3 Анализ основных бизнес-процессов компании

Любая система может быть представлена либо в текстовом, либо в графическом виде.

Бизнес-процесс (БП) – это устойчивая, целенаправленная совокупность взаимосвязанных видов бизнес-операций (БО), направленных на достижение какой-либо определенной цели, с указанием начала и конца, точным определением входов, выходов, механизмов исполнения и управления.

Операция – функция (элемент процесса), которая, при выбранном уровне детализации описания, не делится на составные элементы или «простейшие» работы.

Вход БП – объект бизнес-процесса, взаимодействующий с внешними бизнес-процессами и получающий от них информацию/материальные ресурсы.

Выход БП – объект бизнес-процесса, взаимодействующий с внешними бизнес-процессами и передающий им информацию/материальные ресурсы, являющиеся результатом выполнения бизнес-процесса.

Моделирование бизнес-процесса – процесс отражения субъективного видения потока работ в виде формальной модели. Структуризация общего бизнес-процесса предприятия начинается с построения общей структурной схемы бизнес-процесса организации. При помощи IDEF-технологий, в частности, IDEF0, описание системы представлено в виде модели как совокупность взаимодействующих работ или функций.

Процесс моделирования какой-либо системы в IDEF0 начинается с определения контекста, т.е. наиболее абстрактного (общего) уровня описания системы в целом. В контекст входит определение субъекта моделирования, цели и точки зрения на модель.

В данной работе концептуальная диаграмма A0 представляет из себя функцию «Деятельность предприятия в области продажи и учета топлива».

Цель построения модели – моделирование создаваемых бизнес-процессов (AS TO BE). Далее необходимо перейти на более низкий уровень декомпозиции модели, согласно функционально-структурному анализу из третьей части

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		21

данной работы.

Таким образом, произведем моделирование основных бизнес процессов предприятия AS IS с использованием программного продукта BPWin и IDEF0 и DFD нотаций.

### 2.3.1 Построение модели AS-IS в нотации IDEF0

Представим контекстную диаграмму деятельности предприятия на рисунке 2.2.

На рисунке 2.3 покажем декомпозицию контекстной диаграммы. Видно, что при декомпозиции контекстной диаграммы получаются 3 бизнес процесса:

- занесение значения заказанного топлива в Excel файл;
- прием оплаты заказа;
- осуществление заправки транспортного средства.

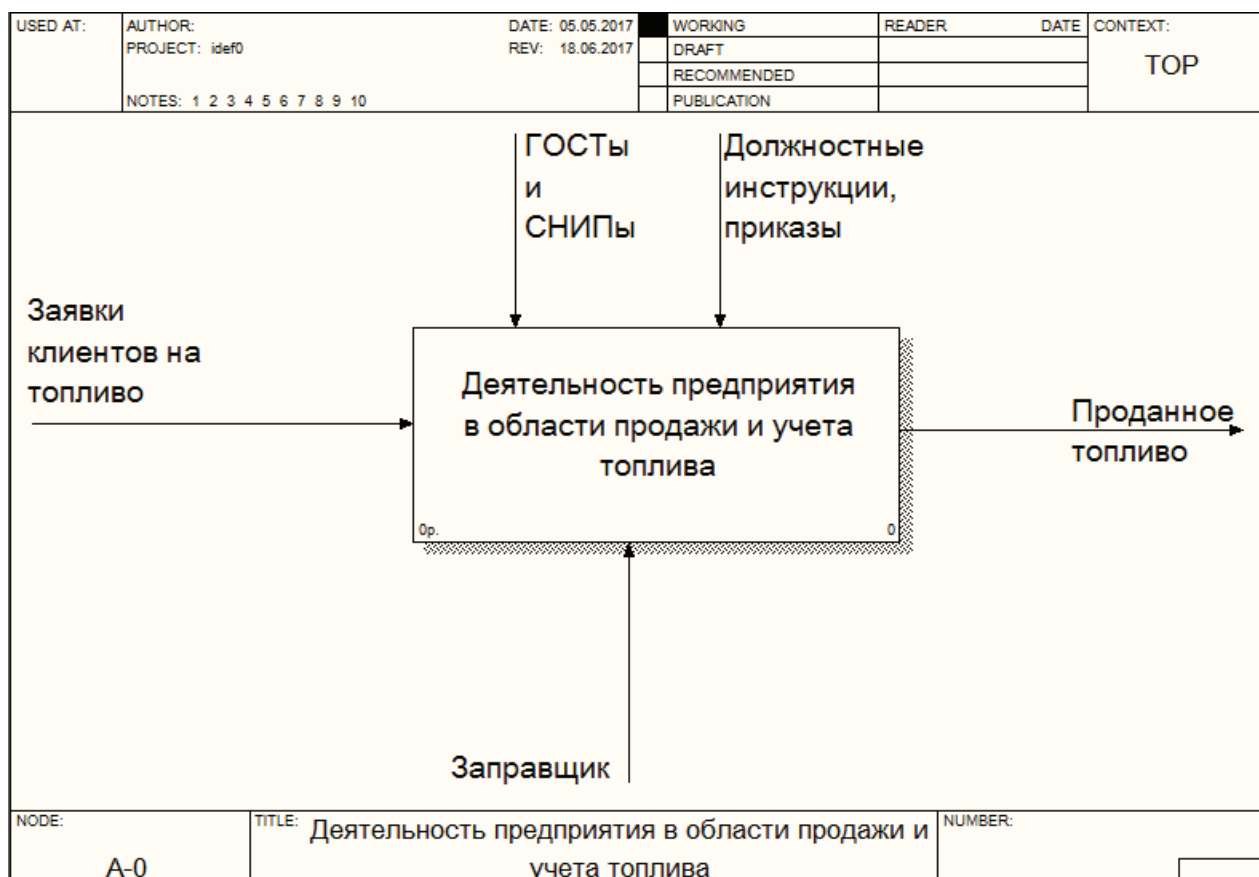


Рисунок 2.2 – Контекстная диаграмма

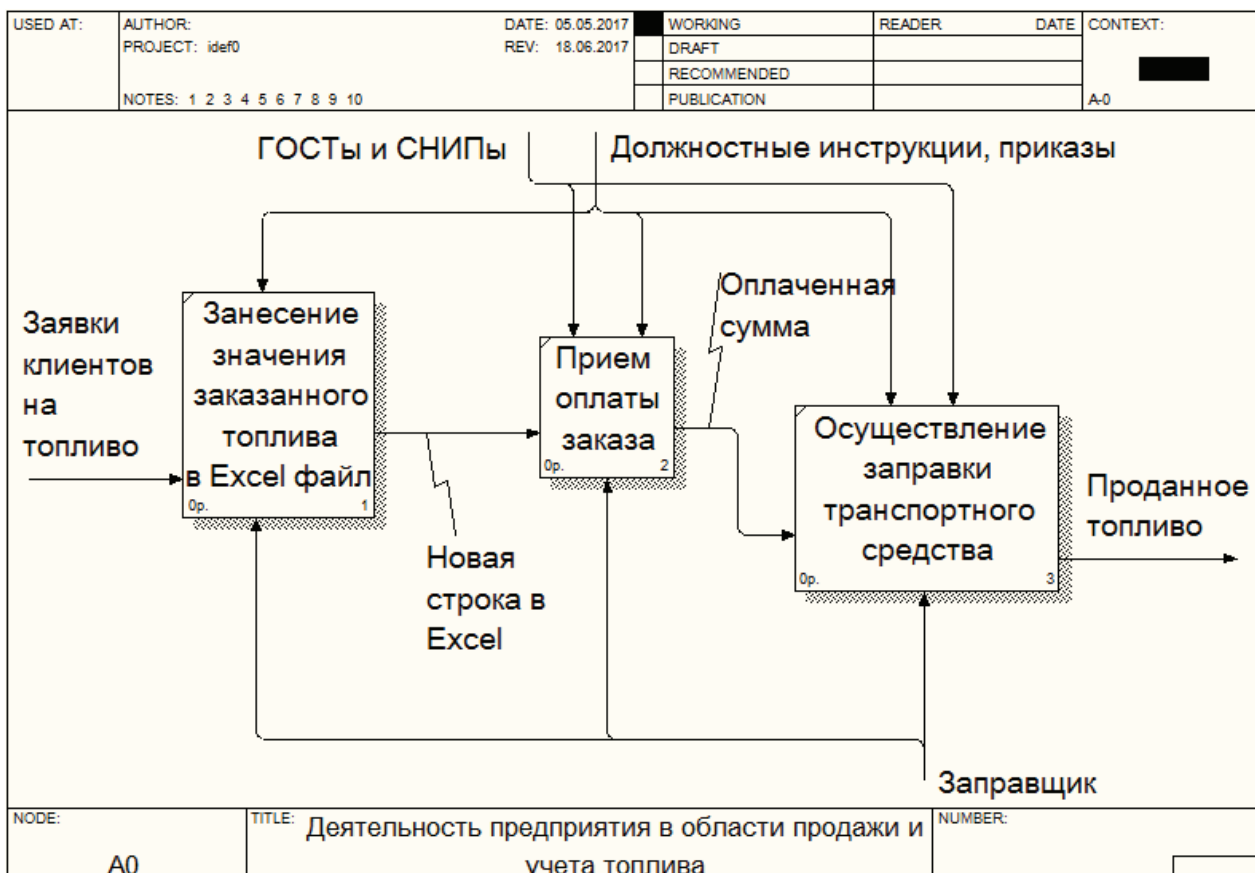


Рисунок 2.3 – Декомпозиция контекстной диаграммы

### 2.3.2 Функциональное моделирование в нотации DFD

Диаграмма потоков данных (DFD-диаграмма) – это граф, на котором показано движение значений данных от их источников через преобразующие их процессы к их потребителям в других объектах. DFD-диаграмма содержит:

- процессы, которые преобразуют данные;
- потоки данных, которые переносят данные;
- активные объекты, которые производят и потребляют данные;
- хранилища данных, которые пассивно хранят данные.

Поток данных соединяет выход объекта (или процесса) со входом другого объекта (или процесса). Он представляет промежуточные данные вычислений. Поток данных изображается в виде стрелки между производителем и потребителем данных.

Представим диаграммы потоков данных (рисунки 2.4 – 2.5).

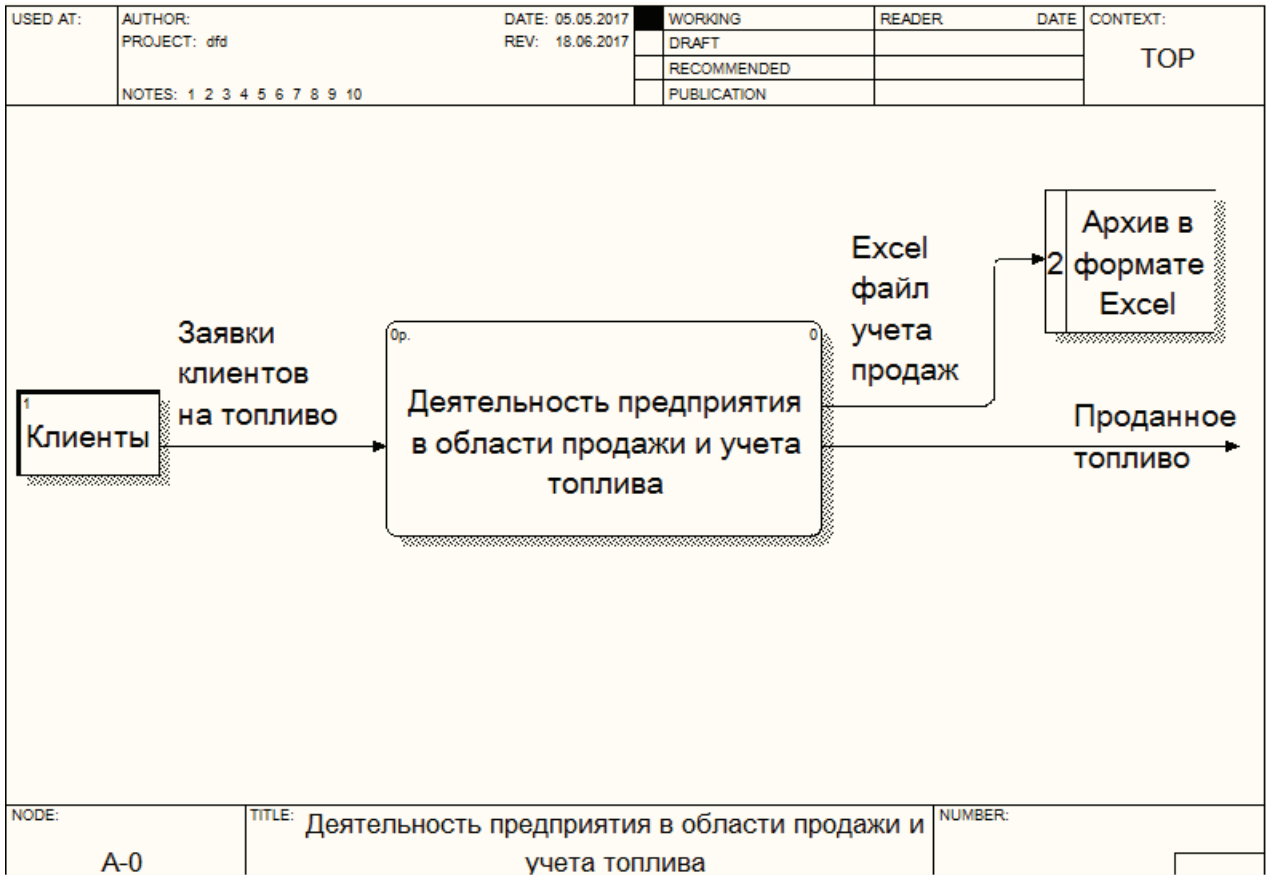


Рисунок 2.4 – Контекстная диаграмма DFD

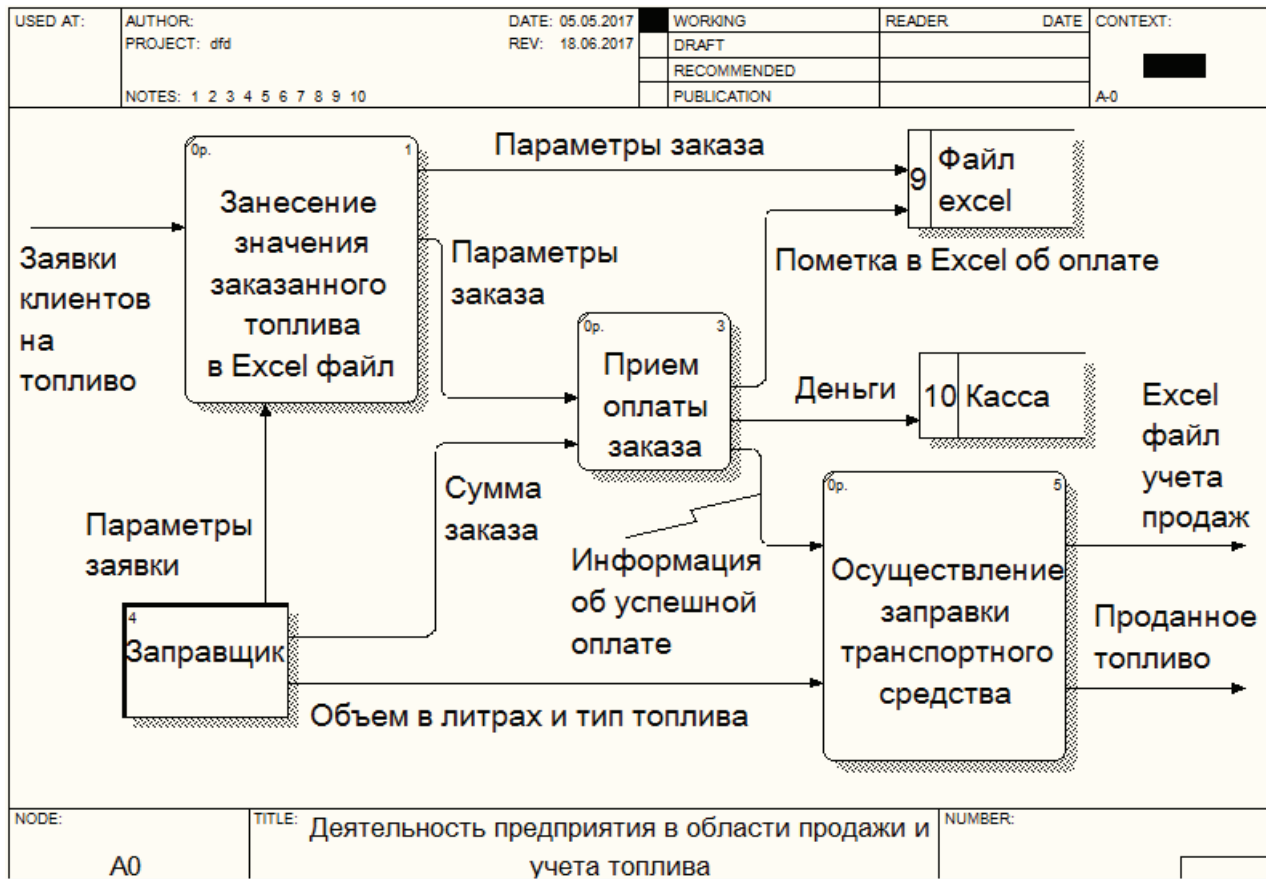


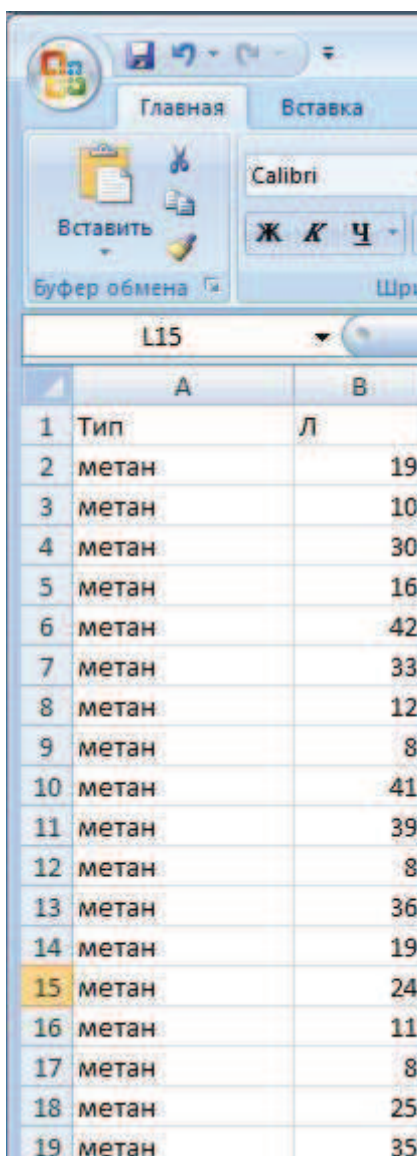
Рисунок 2.5 – Декомпозиция контекстной диаграммы DFD

## 2.4 Обоснование необходимости разработки ИС

Исходя из проведенного моделирования бизнес-процессов предприятия, касающихся продажи и учета топлива можно сделать вывод о том, что эти процессы выполняются недостаточно эффективно.

Для учета проданного топлива используется Excel файл, где каждая новая продажа представляет собой строку в файле. Эти записи сгруппированы по датам (рисунок 2.6).

При этом не ведется учет типов автотранспортных средств, которым продано топливо, времени продажи, что позволило бы анализировать деятельность предприятия по большему числу параметров, а не только по объему проданного топлива, которое можно перевести в выручку.



	A	B
1	Тип	Л
2	метан	19
3	метан	10
4	метан	30
5	метан	16
6	метан	42
7	метан	33
8	метан	12
9	метан	8
10	метан	41
11	метан	39
12	метан	8
13	метан	36
14	метан	19
15	метан	24
16	метан	11
17	метан	8
18	метан	25
19	метан	35

Рисунок 2.6 – Формат Excel файла учета проданного топлива



К тому же данные в этом файле очень сложно обрабатывать для построения статистики продаж, именно поэтому такой анализ не производится.

Стоимость топлива рассчитывается вручную на калькуляторе, что приводит к периодическим ошибкам и недостаткам.

Поэтому предлагается внедрить на предприятии информационную систему, которая будет автоматизировать процесс учета и обработки данных по продажам топлива на АГЗС.

Построим модель проектируемой ИС в рамках нотации DFD (рисунок 2.7-2.9).

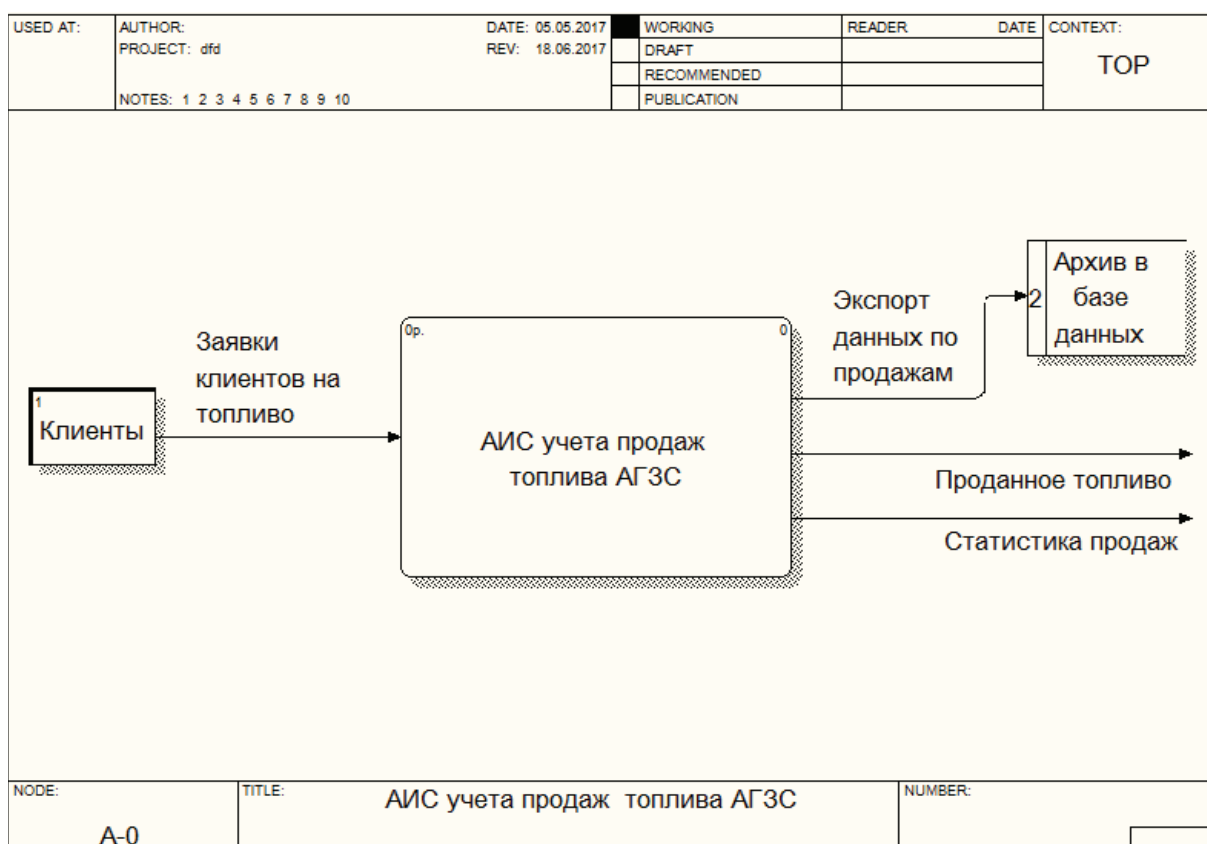


Рисунок 2.7 – Контекстная диаграмма DFD модернизированного БП



Видно, что бизнес процессы претерпели изменения.

Так, вместо занесения записей в Excel происходит автоматическое формирование новой заявки в АИС. Вместо занесения данных вручную, как было раньше, происходит выбор параметров из списка, что требует значительно меньше времени. К тому же, в качестве параметра добавился тип автотранспортного средства.

Автоматическая простановка времени заказа позволит строить статистику и по этому параметру.

С ИС работает не только заправщик, но и начальник заправки, который ведет справочники системы.

В целом, внедрение АИС должно повысить скорость и точность работы заправщика АГЗС.

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		28

## 3 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

### 3.1 Постановка задачи

#### 3.1.1 Основание для разработки

Основанием для разработки служит выделенная потребность предприятия в подобной АИС, а также задание на проектирование.

#### 3.1.2 Назначение

Программный продукт (ПП) предназначен для учета и обработки информации о проданном топливе на АГЗС.

Цели создания:

- автоматизация основных операций учета проданного топлива;
- снижение временных затрат и ошибок на выполнение операций учета;
- создание благоприятного психологического климата при работе с клиентами.

### 3.2 Требования к программе или программному изделию

#### 3.2.1 Требования к функциональным характеристикам

Состав выполняемых функций:

- Регистрация продаж топлива;
- Учет сотрудников;
- Приход топлива, а также его наличие в резервуарах;
- Динамика цен на топливо;
- Автомобили, которые покупали топливо и их типы;
- Клиенты и скидки им;
- Показания приборов для трехчасового отчета;
- Учет выручки;
- Товары в магазине, их приход, расход и наличие в магазине;
- Управление пользователями системы.

Входные данные – вносимые заправщиком и начальником сведения о видах

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		29

топлива, типах автотранспортных средств, конкретном типе авто при заправке, объеме литров топлива, товарах в магазине.

Выходные данные – сумма к оплате, отчет по продажам (часы, период), выручке, ценам на топливо.

### 3.2.2 Требования к видам обеспечения

#### Требования к информационному обеспечению системы

Уровень хранения данных в системе должен быть построен на основе современных СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Средства СУБД, а также средства используемых операционных систем должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Структура базы данных должна быть организована рациональным способом, исключающим единовременную полную выгрузку информации, содержащейся в базе данных системы.

#### Требования к лингвистическому обеспечению

Для создания автоматизированной системы можно применять любые языки программирования высокого уровня 4GL, например: Borland C++, Borland Delphi 7.0, Visual C++, C#, Microsoft FoxPro 9,0 с компилятором для ОС Windows 7/8/10.

#### Требования к программному обеспечению системы

При проектировании и разработке системы необходимо максимально эффективным образом использовать ранее закупленное программное обеспечение.

Используемое при разработке программное обеспечение и библиотеки программных кодов должны иметь широкое распространение, быть общедоступными и использоваться в промышленных масштабах. Базовой программной платформой должна являться операционная система MS Windows.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						30
Изм.	Лист	№ докум.	Подпись	Дата		

### 3.2.3 Требования к надежности

Предусмотреть контроль вводимой информации и блокировку некорректных действий пользователя при работе с системой.

Обеспечить корректное завершение вычислений с соответствующей диагностикой при превышении имеющихся вычислительных ресурсов.

### 3.2.4 Требования к составу и параметрам технических средств

Для нормальной работы с ИС рабочие места пользователей должен иметь характеристики не ниже указанных:

- операционная система Windows 7 и выше;
- установленный .Net Framework 4.0 и выше;
- установленная СУБД MySQL 5.5 и выше;
- ОЗУ 2048 Мб и выше;
- сетевая карта.

### 3.2.5 Требования к информационной и программной совместимости

Система должна работать под управлением операционной системы Windows 7 и выше.

### 3.2.6 Требования к программной документации

Система может использоваться без программной документации.

## 3.3 Анализ существующего программного обеспечения на рынке

В настоящее время на рынке присутствуют различные программные средства для автоматизации АЗС:

- Система управления АЗС «БУК TS-G»;
- Система «GasKit».

### Система управления АЗС «БУК TS-G»

Важнейшим направлением деятельности этой компании является поставка комплексных систем автоматизации и коммерческого учета на АЗС. Разработчик предлагаемой системы – компания «Нефтепродукттехника»,

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						31
Изм.	Лист	№ докум.	Подпись	Дата		

является разработчиком и производителем системы автоматизации АЗС (далее Система) «БУК TS-G», которая сертифицирована и допущена к использованию на всей территории Российской Федерации (Госреестр средств измерений, № 18111-05 от 08.09.2005 года).

На сегодняшний день Система «БУК TS-G» эксплуатируется более чем на 1400 АЗС в России и странах СНГ.

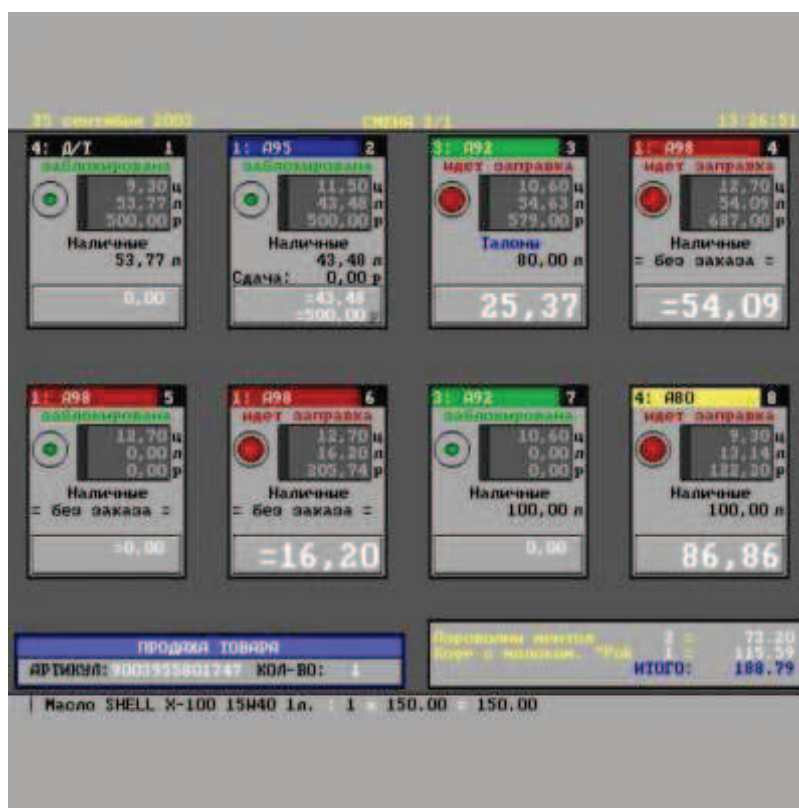


Рисунок 3.1 – Система АЗС «БУК TS-G»

Система «БУК TS-G» разработана с учетом всех современных требований к системам автоматизации АЗС, при этом ориентирована как на небольшие АЗС, так и на крупные заправочные комплексы с организацией нескольких рабочих мест операторов и администратора АЗС. Система поддерживает работу с кассами, уровнемерами, обеспечивает прием различных видов оплаты, в том числе осуществляет отпуск топлива по талонам со штрих-кодом, магнитным, смарт- и электронным картам (платежным и дисконтным), продажу товаров и услуг, организацию пересылки информации о движении топлива и товаров на АЗС по модему на удаленный компьютер и многое другое.

Одной из дополнительных опций, поставляемой с Системой «БУК TS-G»,

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		32

является система безналичных расчетов на АЗС с применением электронных карт. В качестве носителя информации используются электронные карты Touch Memory американской фирмы «Dallas Semiconductor Co.», выполненные в виде «брелка с таблеткой». На основе этой технологии они поставляют безоператорные АЗС.

## Система «GasKit»

### Особенности системы «GasKit»:

- обзор и регистрация в базе данных значений уровня топлива, температуры и плотности, получаемых с заданным периодом от систем измерения;
- подробный учет движения нефтепродуктов и сопутствующих товаров с формированием отчетных документов;
- пользовательский интерфейс оптимизирован для работы с сенсорным экраном;
- обслуживание корзины с любого рабочего места;
- несколько видов топлива в одной корзине;
- очередь заказов, позволяющая отпускать топливо следующему покупателю до окончания расчета с предыдущим;

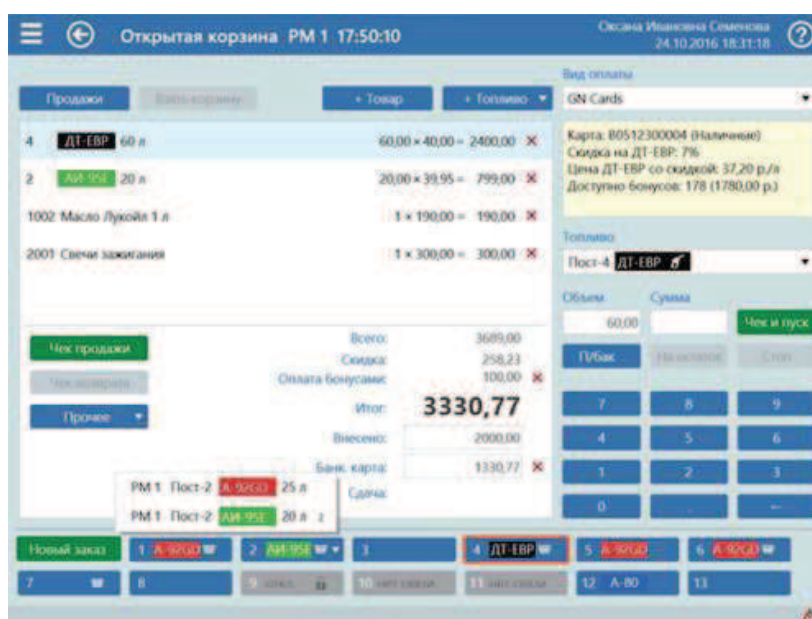


Рисунок 3.2 – Система «GasKit»



Система «GasKit» имеет гибкую структуру видов оплат и скидок:

- наличные и безналичные формы оплаты, расчет скидок по заданным условиям, печать соответствующих фискальных и товарных чеков;
- смешанная оплата, когда в одном чеке регистрируется платеж наличными, банковскими картами и бонусами;
- начисление скидок и бонусов может зависеть от суммы чека, истории покупок, количества товаров, вида товара, времени и множества других факторов.

ПО имеет широкие возможности по настройке системы, позволяющие учесть особенности бизнеса. Модульность системы позволяет конфигурировать ее в соответствии с пожеланиями заказчиков:

- настройка прав различных категорий пользователей системы «GasKit» на просмотр информационных таблиц, редактирования данных, выполнения различных функций;
- автоматический поиск технологического и торгового оборудования;
- проверка введенных значений параметров в конфигурации АЗС и отсутствие противоречий между ними.

Рассмотренное ПО обладает внушительным функционалом, выходящим за рамки требований рассматриваемого предприятия. К тому же программные продукты являются платными. С учетом того, что предприятие предъявляет не такие значительные требования к функционалу, самостоятельная разработка системы является обоснованной. По мере необходимости такая система в будущем может совершенствоваться путем добавления дополнительных модулей.

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		34

### 3.4 Выбор и обоснование инструментальных средств для решения поставленной задачи

#### 3.4.1 Выбор СУБД

Приведем в таблице 3.1 краткое описание СУБД и их основные параметры.

Таблица 3.1 – Краткое сравнение СУБД

База данных	Описание
SQLite	Повсеместно применяется. Состоит из одного файла, однако во время работы с БД могут создаваться и другие файлы. Библиотеки для доступа к БД есть на всех современных языках программирования и запускаются на любой платформе. Бесплатная БД.
MS ACCESS	Является частью MS Office. Как и в SQLite, читать БД может несколько клиентов, писать – один. БД имеет закрытый программный код, не понятно соглашение. С одной стороны, пользоваться БД можно в случае если имеется лицензия на MS Office. В то же время использовать ACCESS без установленного MS Office можно при помощи поставщика данных ODBC.
SQL Server Compact	Эта БД, как SQLite, хранит все данные в одном файле, однако для работы требуется несколько неуправляемых библиотек и провайдер.
MS SQL	Серверная БД от Microsoft. При написании скриптов применяется Transact-SQL. В Full-версии поддерживает OLAP и присутствуют функции data mining (для добычи данных и сбора статистики). Среди редакций БД есть бесплатная – SQL Express.
Oracle	Одна из старейших серверных БД. Использует PL-SQL, а также обязательно нужно применять sequence-ы к инкрементируемым полям. Лицензия на БД стоит больших денег, но, как и у Microsoft, есть бесплатная версия – Oracle XE.
Postgre	Серверная БД уровня предприятия с открытым кодом. Не имеет единого разработчика. Быстрая и полнофункциональная БД, но для оптимизации может потребовать доработок.
MySQL	MySQL сочетает в себе несколько алгоритмов хранения данных. Данное обстоятельство, совместно с простотой применения сыграло важную роль в популяризации этой БД. В настоящее время в большинстве случаев эта БД применяется в веб разработке для хранения данных на хостингах.
FireBird	Бесплатный клон БД InterBase от Borland. Код закрыт, однако ограничений на применение нет.

В качестве СУБД была выбрана MySQL поскольку она самая распространенная полноценная серверная СУБД. Также MySQL очень функциональная, свободно распространяемая СУБД, которая успешно работает

с различными сайтами и веб приложениями. Обучиться использованию этой СУБД также довольно просто.

### 3.4.2 Выбор языка программирования и среды разработки

Кратко расскажем о наиболее важных из существующих в настоящий момент языков программирования.

#### C

C является одним из первых языков программирования, который стал популярными и используется до сих пор. Этот язык создан в 1972 году Деннисом Ритчи в Bell labs и является основой для многих других языков: C#, JavaScript, C++, Java, Perl. Этот язык обычно первым преподаётся в ВУЗах, так как помогает лучше понимать многие составляющие технологии программирования.

Многие популярные ресурсы в интернете помогают бесплатно освоить язык C.

#### C++

Для своего времени язык C был хорошим, однако с течением времени ему стало не хватать объектно-ориентированности. Потребовалось реализовать эту потребность. Так, в 1983 г. был разработан более мощный язык, который получил название C++. Его часто называют объектно-ориентированной версией языка C. C++ – один из самых часто используемых императивных языков в мире. Разработчики его применяют для создания разного рода приложений: браузеры Mozilla Firefox, Google Chrome, музыкальный проигрыватель Winamp, а также продукты Adobe – все это и много другое написано на языке C++.

Также C++ используют для разработки игр и приложений для смартфонов. На рынке труда разработчики C++ востребованы и число вакансий растёт.

#### Java

Язык Java сочетает в себе возможности C и C++ и является мощным языком создания кроссплатформенных приложений. Java, стал первым полностью объектно-ориентированным языком и функционирует по принципу «Однажды

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		36

написал, используй везде». Исходный код Java можно скомпилировать однажды и затем запускать на любом устройстве. Требование к устройству – наличие установленной виртуальной машины Java (JVM).

## C#

C# является представителем семейства языков программирования от компании Microsoft. Он создан в 2000 году в качестве составной части платформы Microsoft .Net Framework. C# весьма близок к Java в использовании, хотя есть мнение о том, что C# сочетает в себе надёжность C++ вместе с продвинутыми характеристиками языка Java.

На C# можно создавать почти все типы приложений, к тому же его можно использовать для написания ПО для большинства современных платформ.

## PHP

Многие вполне обоснованно полагают, что PHP является самым мощным серверным языком, который может выполнять большой объем работы при незначительном количестве строк кода. PHP широко применяется для создания различных CMS, например, Joomla, Drupal, WordPress и др., а также для разработки статических и динамических веб-сайтов. PHP представляет собой язык с открытым кодом. Это означает, что в открытом доступе существуют тысячи модулей и решений, которые могут быть легко модифицированы для нужных программисту функций.

## Perl

Создателем этого языка является Ларри Уолл, который разработал Perl в 1987 году для упрощения обработки различных отчетов. Неофициальный бэкроним языка – Practical Extraction and Report Language. Perl представляет собой высокоуровневый современный язык программирования общего назначения, который, в принципе, может заменить любой другой язык программирования. В данное семейство входят Perl 5, Perl 6. Языки семейства Perl позаимствовали черты таких языков, как C, AWK, sh, sed.

## Python

Создателем Python является Гвидо ван Россум, который создал его в 1991

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						37
Изм.	Лист	№ докум.	Подпись	Дата		

году. Python представляет собой высокоуровневый язык программирования. Он очень прост для изучения благодаря своему минимализму, понятному синтаксису и хорошей читабельности.

## Ruby

Ruby очень похож на Python – это простые в понимании и читабельные языки программирования, которые используются в основном для разработки веб-приложений. Ruby разработан в 1995 году Юкихио Мацумото. Обычно чистый Ruby не используется, а используется его фреймворк RubyOnRails. Это мощный фреймворк веб-разработки. На нем написаны многие сайты, включая Yammer, Shopify, Groupon, Github, Scribd.

В качестве языка программирования выбран C#, которому соответствует среда разработки Visual Studio. Этот выбор обусловлен тем, что C# фактически интегрировал в себя все плюсы языков Java и C++, а также имеет такие дополнительные особенности, как автоматический сборщик мусора, продвинутую систему поддержки разработки кода, встроенную в IDE, множество удобных и полезных типов данных, значительно ускоряющих процесс разработки. К тому же IDE Visual C# Express является бесплатной средой разработки приложений на C#.

## 3.5 Разработка инфологической и даталогической моделей данных

### 3.5.1 Инфологическая модель

Концептуальный уровень – центральное управляющее звено, здесь база данных представлена в наиболее общем виде, который объединяет данные, используемые всеми приложениями, работающими с данной базой данных. Фактически концептуальный уровень отражает обобщенную модель предметной области (объектов реального мира), для которой создавалась база данных. Как любая модель, концептуальная модель отражает только существенные, с точки зрения обработки, особенности объектов реального мира.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		38

Опишем инфологическую модель.

Имеется предприятие АГЗС, которое оказывает услуги по заправке автотранспортных средств метаном, бензином, а также продаже товаров.

Автомобили могут быть разных типов: грузовые, легковые.

У каждого клиента для его идентификации должен быть определенный номер, по которому система сможет отслеживать его покупки.

В результате инфологического моделирования выделим такие сущности:

- хранилище топлива;
- топливо;
- цены на топливо;
- автомобили;
- типы автомобилей;
- пользователи;
- должности предприятия;
- заказы;
- автозаправочные станции;
- приход топлива;
- клиенты;
- типы клиентов;
- карты клиентов;
- скидки;
- выручка;
- показания приборов учета;
- настройки;
- продукты;
- продукты на складе;
- приход продуктов;
- продажи продуктов.

### 3.5.2 Даталогическая модель базы данных

Построение даталогической модели базы данных – это следующий этап проектирования базы данных. Исходными данными для даталогического проектирования является инфологическая модель предметной области.

В результате должна быть получена логическая структура базы данных, описанная в терминах выбранной модели данных на основе физических записей. Даталогическая модель представлена на рисунке 3.3. Она построена в CASE-средстве проектирования Allfusion ERWin Data Modeler 7.

При переходе к даталогической модели следует помнить, что инфологическая модель включает в себя всю информацию о предметной области, необходимую и достаточную для проектирования баз данных.

Преобразование связи 1:М. Связь реализуется копированием первичного ключа из реляционного отношения на стороне «один» в реляционное отношение на стороне «много», из главного отношения в подчинённое. Новому появившемуся атрибуту присваивается уникальное в пределах отношения имя. Этот вновь появившийся атрибут помечается как внешний ключ.

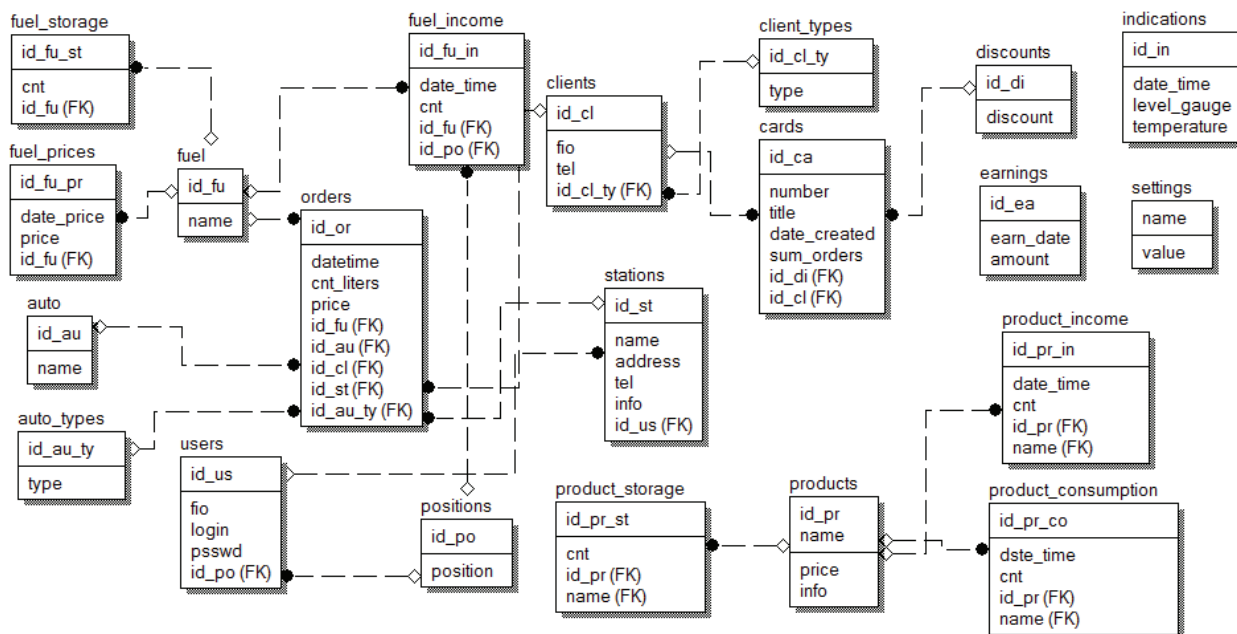


Рисунок 3.3 – Даталогическая модель базы данных

Преобразование связи 1:1. В даталогической модели связь 1:1 может иметь разную опциональность. От этого зависит ее отображение в схеме БД. Внешний

ключ создаётся копированием первичного ключа из главного отношения в подчинённое.

При построении даталогической модели данных следует учитывать и принципы нормализации до третьей нормальной формы:

Первая нормальная форма – каждая запись в базе данных представляет один экземпляр сущности. Атомарность – поля не имеют дубликатов в каждой записи, и каждое поле содержит только одно значение.

Вторая нормальная форма – поля с не первичным ключом не должны быть зависимы от первичного ключа. Следование второй нормальной форме – это вопрос нахождения данных, которые часто дублируются в записях таблицы и которые могут принадлежать другой сущности.

Третья нормальная форма – не может быть транзитивных зависимостей между полями в таблице. Транзитивные зависимости между полями базы данных существует тогда, когда значения не ключевых полей зависят от значений других не ключевых полей.

Приведённая схема логической структуры реляционной базы данных соответствует, как минимум, третьей нормальной форме, поскольку во всех отношениях атрибуты атомарные, первичные ключи не составные и отсутствуют транзитивные зависимости между не ключевыми атрибутами и первичными ключами.

Видно, что для пользователей системы добавлена новая таблица positions, чтобы исключить дублирование должностей в таблице пользователей ИС.

### 3.5.3 Физическая модель данных

Физическая модель данных должна быть построена в рамках типов данных, применяемых в СУБД MySQL (рисунок 3.4).

Скрипт инициализации БД представлен в приложении А. Он сгенерирован автоматически в CASE-средстве проектирования Allfusion ERWin Data Modeler 7.

Скрипты инициализации БД представлены в приложении А.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		41



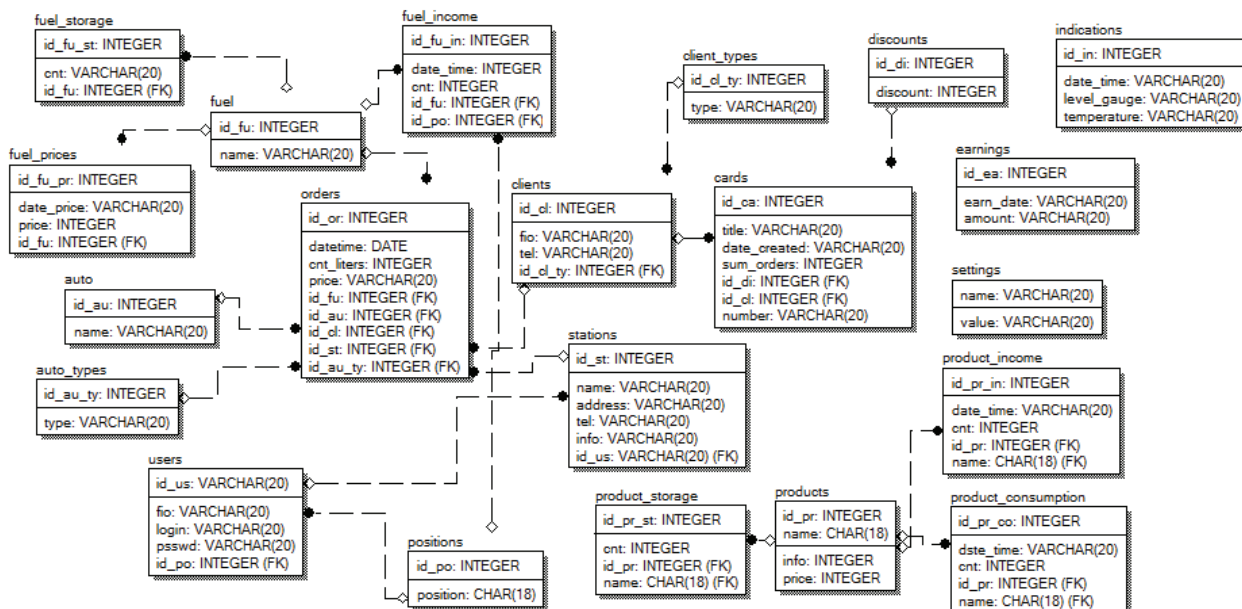


Рисунок 3.4 – Физическая модель данных

### 3.6 Разработка интерфейса информационной системы

В широком смысле интерфейс представляет собой определенную стандартами границу между независимыми объектами, обменивающимися между собой некоторой информацией. Интерфейс задает параметры объектов, а также их процедуры и характеристики взаимодействия.

В понятие пользовательского интерфейса компьютерной системы входят следующие составляющие:

- графическая среда – картинка на экране;
- набор управляющих элементов пользовательского интерфейса и их расположение на экране;
- технологии взаимодействия пользователя с системой.

Управляющие элементы пользовательского интерфейса – это графические элементы (кнопки, списки, диалоговые окна и т.п.), которые позволяют осуществлять какие-либо действия с компьютерной системой (например, выбирать пункты и свойства объектов).

Основные требования к пользовательскому интерфейсу:

- функциональность (соответствие задачам пользователя);
- соответствие технологии;

- понятность и логичность;
- обеспечение высокой скорости работы пользователя;
- обеспечение защиты от человеческих ошибок;
- быстрое обучение пользователя;
- субъективная удовлетворенность пользователя.

Для того, чтобы достичь выполнения указанных требований к интерфейсу, нужно соблюдать ряд правил:

1) для повышения скорости выполнения работы элементы управления следует делать заметными и понятными;

2) для уменьшения количества ошибок пользователя:

- не следует делать кнопки слишком маленького размера: на такую кнопку сложно попасть (кнопкой называется элемент управления, всё взаимодействие пользователя с которым ограничивается одним действием – нажатием);
- нужно сделать так, чтобы пользователю было трудно нажать не на ту кнопку (добиться этого можно либо изменением состояния кнопки при наведении на неё курсором, либо с помощью визуальных пауз между кнопками);
- не следует делать опасные для пользователя кнопки кнопками по умолчанию;
- лучше показывать границы диапазона вводимых пользователем значений;
- следует избегать создания элементов управления, функции которых меняются в зависимости от контекста.

На основе общих требований к пользовательскому интерфейсу сформирована система требований к его элементам управления. Требования к названию (тексту) элементов управления:

- название элемента должно отражать его функцию;
- названия элементов должны быть краткими, но понятными

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						43
Изм.	Лист	№ докум.	Подпись	Дата		

пользователю;

- наиболее значимое слово должно стоять в названии элемента первым;
- для названия элемента, запускающего действие, целесообразно использовать глагол в форме инфинитива;
- если элемент меню служит для запуска окна с продолжением диалога, то в конце его названия следует ставить многоточие;
- пиктограммами следует снабжать только самые важные элементы меню.

Требования к расположению элементов управления:

- элементы меню следует группировать;
- группы следует разделять разделительными полосками либо «визуальными паузами»;
- часто используемые элементы целесообразно располагать в левой верхней части экрана, редко используемые – в правой нижней части;
- командные кнопки, управляющие окном (например, «Ок», «Отмена», «Применить», «Закрыть»), должны быть расположены либо внизу окна, либо в правой его части (т.е. в той части окна, которая сканируется взглядом в последнюю очередь);
- хорошо, если диалоговое окно читается, как текст: один элемент управления однозначно преобразовывается во фрагмент предложения, а единая группа элементов – в целое предложение.

Требования к оформлению переключателей и флажков:

- не следует нарушать сложившуюся традицию: переключатели выглядят как квадраты, а флажки – как кружки;
- и переключатели, и флажки желательно расставлять по вертикали, поскольку это значительно ускоряет поиск нужного элемента;

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		44

- флажки всегда должны находиться в рамке группировки, а для переключателей это необязательно;
- в группе флажков как минимум один флажок должен быть проставлен по умолчанию;
- каждая подпись переключателя и флажки должны однозначно показывать эффект от выбора соответствующего элемента;
- поскольку флажки и переключатели не вызывают немедленного действия, формулировать подписи к ним лучше всего в форме существительных, хотя возможно использование глаголов;
- подписи к стоящим параллельно кнопкам лучше стараться делать примерно одинаковой длины;
- все подписи лучше делать позитивными (т.е. не содержащими отрицания);
- не следует повторять в подписях одни и те же слова, меняя только окончания подписей (например, «Показывать пробелы» и «Показывать табуляции») – лучше перенести повторяющееся слово в рамку группировки;
- если подпись не помещается в одну строку, то индикатор кнопки (кружок или квадрат) следует выравнивать по первой строке подписи;
- при необходимости заблокировать элемент переключателя или флажки, желательно визуально ослаблять не только индикатор кнопки (квадрат или круг), но и подпись к нему.

Представим разработанные макеты шаблонов форм входной и выходной информации для проектируемого приложения (с помощью встроенного редактора в Microsoft Visual Studio). Макеты представлены в приложении Б.

### 3.7 Разработка программных модулей

Представим таблицу описания функций модулей (таблица 3.2).

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						45
Изм.	Лист	№ докум.	Подпись	Дата		

Таблица 3.2 – Описание функций модулей

№ п/п	Наименование модуля	Функции модуля
2	Модуль инициализации	Содержит функции по первоначальной настройке ИС. Инициализацию вспомогательных подмодулей.
3	Модуль системной логики	Содержит в себе иерархию подчиняющихся форм приложения, что обеспечивает логичную и интуитивно понятную программную навигацию
4	Модуль подключения к БД	Отвечает за обеспечение связи приложения с БД MySQL.
5	Модуль взаимодействия с БД	Отвечает за выполнение запросов к БД и предоставление для других модулей ИС унифицированных процедур для выполнения основных операций над данными: чтение, запись, изменение, удаление.
6	Модуль пользовательского интерфейса	Содержит функции, включающие: Формирование пользовательского интерфейса согласно хранимым шаблонам

Вообще, данное разделение больше логическое, чем физическое, поскольку в реальности присутствуют несколько десятков файлов, каждый из которых отвечает за определенный набор специфических функций. Однако все эти файлы могут быть разделены на логические группы, представленные в таблице выше.

В качестве языка программирования выбран C# – соответственно, средой программирования является MS Visual Studio.

Теперь перейдем к физическому представлению модулей. ИС состоит из 10 модулей (рисунок 3.5). Спецификация всех модулей ИС приведена в таблице 3.3.

Таблица 3.3 – Спецификация модулей

Название модуля	Назначение модуля	Вызывающий модуль
Program.cs	Инициализация приложения	При запуске программы
auth.cs	Главная страница	Program.cs

main.cs	Авторизация пользователей	auth.cs
products.cs	Операции с продуктами	main.cs
contacts.cs	Операции с клиентами	main.cs
actions.cs	Операции с приходами и продажами топлива	main.cs
FormTpl	Редактирование справочников	main.cs
Excel.cs	Вывод данных в excel	analysis.cs
analysis.cs	Подготовка и анализ данных	main.cs
Report.cs	Отчет по продажам	main.cs

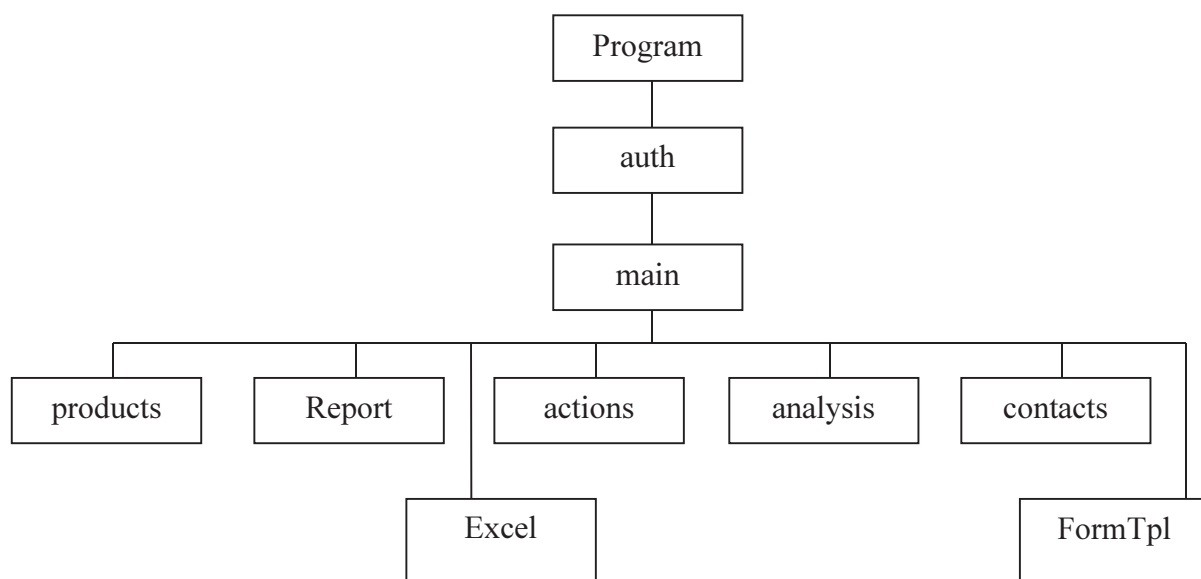


Рисунок 3.5 – Иерархическая схема модулей ИС

Модуль Program.cs – является точкой входа приложения и служит для первоначальной инициализации приложения в целом.

Модуль auth.cs – модуль авторизации, который позволяет ограничить попытки входа в приложение до 3-х при неправильном вводе пароля, а также производит аутентификацию и первоначальную установку прав доступа для вновь идентифицированного пользователя.

Модуль main.cs – главное окно приложения, отображает текущие продажи топлива, а также содержит набор функций логики приложения.

Модуль products.cs – просмотр и редактирование продуктов магазина.

Модуль contacts.cs – просмотр и редактирование контактов. Отметим, что под контактом понимается клиент предприятия и связанные с ним данные.

Модуль actions.cs – выполнение операций по приходу и продажам топлива.

Модуль FormTrl – создает и отображает справочники с возможностью их редактирования.

Модуль Excel.cs – служит для распечатки таблиц формы в файл Excel.

Модуль analysis.cs – подготовка данных для анализа, запуск анализа, подготовка и отображение результатов анализа.

Модуль report.cs – подготовка отчета по продажам за период с возможностью выгрузки в Excel.

### Описание программных модулей

Отметим вначале, что существуют стандартные операции над данными, так называемый CRUD:

- создание;
- чтение;
- изменение;
- удаление.

Данные для ИС хранятся в базе данных. Интерфейс взаимодействия с БД предоставляет СУБД MySQL.

Наша задача – сформировать некие стандартизированные механизмы управления потоками данных в приложении, чтобы, во-первых, не дублировать код, во-вторых, обеспечить логичную цепочку передачи сообщений по каналам управления.

В рамках разработки ПО одним из основных моментов является взаимодействие с базой данных. Были разработаны универсальные функции выполнения запросов к БД:

- SELECT;
- INSERT;
- другие запросы.

Вначале следует установить соединение с БД:

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		48

```

public MySqlConnection getConnection() // Получить соединение с БД
{
    try
    {
        if (cn == null || cn.State != ConnectionState.Open)
        {
            cn = new MySqlConnection();
            cn.ConnectionString = connString;
            cn.Open();
            connected = true;
            return cn;
        }
        else
        return cn;
    }
    catch(MySqlConnection ex)
    {
        if(DEBUG)
            MessageBox.Show("Ошибка подключения к БД:\r\n"
+ ex.Message);
        return conn;
    }
}

```

Ниже представлены функции выполнения MYSQL запросов:

```

public bool MySqlSelect(String MySql, ref List<List<String>>
MySqlresult) // Выполнить запрос к БД
{
    try
    {
        conn = getConnection();
        MySqlCommand myCommand = new
MySqlCommand(MySql, conn);
        MySqlDataReader rd = myCommand.ExecuteReader();
        MySqlresult.Clear();
        while(rd.Read())
        {
            List<String> row = new List<String>();
            for (int i = 0; i < rd.FieldCount; i++)
            {
                row.Add(rd.GetValue(i).ToString());
            }
            MySqlresult.Add(row);
        }
        closeConnection(conn);
    }
    catch (Exception ex)
    {
        if (DEBUG)
            MessageBox.Show("Ошибка выполнения MYSQL
запроса:\r\n"+MySql+"\r\n"+ex.Message);
        return false;
    }
}

```

					<b>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</b>	<i>Лист</i>
						49
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		



```
    }  
    return true;  
}
```

Для выполнения запросов используется компонент `MySqlDataReader`.

Результат записывается в коллекцию `List<List<String>>` `MySqlresult`, которая представляет собой список списков строк и позволяет хранить возвращенную в результате запроса таблицу.

Листинги некоторых программных модулей представлены в приложении В. Таким образом, рассказаны основные особенности программной реализации ИС.

### 3.8 Тестирование разработанных компонентов информационной системы

При запуске приложение появляется окно авторизации (рисунок 3.6). В нем требуется ввести параметры подключения к СУБД, логин пользователя и пароль.

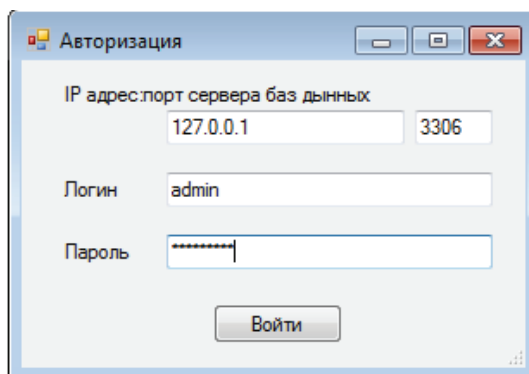


Рисунок 3.6 – Авторизация пользователей

На данной форме пользователю предлагается ввести ip-адрес сервера, порт сервера, логин и пароль пользователя.

После успешной авторизации открывается основная форма (рисунок 3.7), на которой отображаются текущие продажи топлива.

Дата заказа	Продано литров	Цена	Топливо	Авто	Тип авто	Клиент
01.06.2017 23:24	50	624.5	Метан	A983AA 174 RUS	Грузовой	Муравьев Сергей Алексеевич
01.06.2017 22:58	20	249.8	Метан	A239AA 174 RUS	Легковой	Слуцкий Анатолий Ильич
01.06.2017 21:58	80	999.2	Метан	A934AA 174 RUS	Грузовой	Кононенко Владимир Константин...
01.06.2017 21:46	70	874.3	Метан	A185AA 174 RUS	Грузовой	Гагарин Андрей Петрович
01.06.2017 21:37	40	499.6	Метан	A464AA 174 RUS	Легковой	Балобеев Андрей Викторович
01.06.2017 21:20	70	874.3	Метан	A111AA 174 RUS	Грузовой	Бочаров Александр Вячеславович
01.06.2017 21:10	70	874.3	Метан	A990AA 174 RUS	Грузовой	Канзачаков Александр Владимир...
01.06.2017 20:26	80	999.2	Метан	A530AA 174 RUS	Грузовой	Салькова Алия Кайратовна
01.06.2017 20:17	50	624.5	Метан	A231AA 174 RUS	Грузовой	Фрик Екатерина Олеговна
01.06.2017 19:23	50	624.5	Метан	A581AA 174 RUS	Грузовой	Качаев Эльгиз Идрисович
01.06.2017 18:52	10	124.9	Метан	A475AA 174 RUS	Легковой	Кассюра Евгений Владимирович
01.06.2017 16:34	40	499.6	Метан	A904AA 174 RUS	Легковой	Поддубная Светлана Николаевна
01.06.2017 16:24	70	874.3	Метан	A756AA 174 RUS	Грузовой	Наумова Анастасия Леонидовна
01.06.2017 15:53	80	999.2	Метан	A531AA 174 RUS	Грузовой	Дорофеева Ольга Андреевна
01.06.2017 15:10	40	499.6	Метан	A833AA 174 RUS	Легковой	Котовская Анна Николаевна
01.06.2017 14:42	20	249.8	Метан	A326AA 174 RUS	Легковой	Хотяков Виктор Владимирович

Рисунок 3.7 – Отображение продаж топлива

Таблица отображает актуальную информацию о продажах топлива. Указываются такие параметры, как:

- дата заказа;
- количество проданных литров;
- цена заказа, которая автоматически рассчитывается системой на основе справочника цен за топливо;
- вид проданного топлива;
- номер автомобиля;
- тип автомобиля;
- ФИО клиента.

Также пользователю доступны пункты меню «Справочники» и «Операции» (рисунок 3.8).

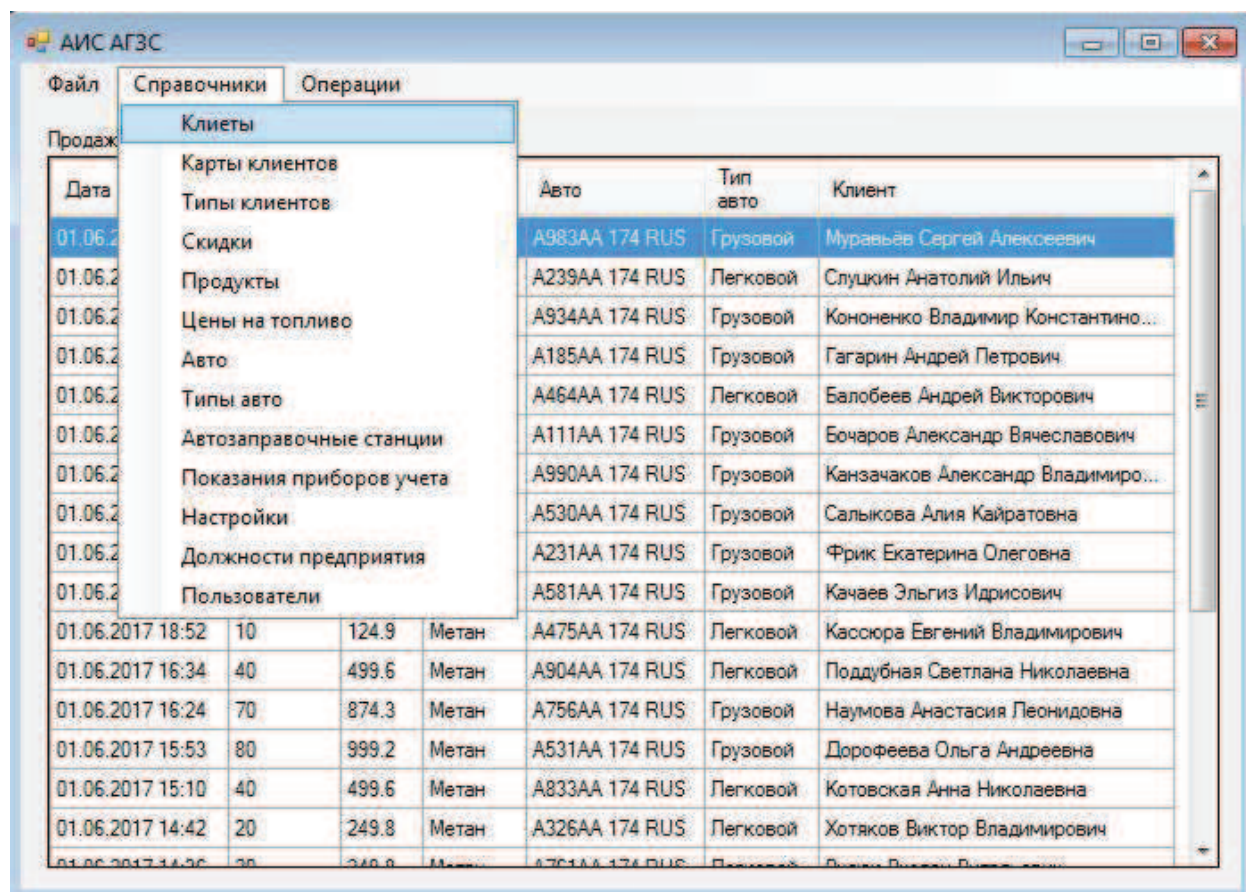


Рисунок 3.8 – Справочники системы

Все справочники представляют собой таблицы, над которыми можно выполнять операции:

- просмотра записей;
- добавление записей;
- редактирования записей;
- удаления записей.

Для того, чтобы показать внешний вид типового справочника разработанной системы, покажем справочник «Клиенты» (рисунок 3.9).

Клиент	Тип клиента	Телефон
Муравьев Сергей Алексеевич	Новый	8-911-949-90-09
Слуцкий Анатолий Ильич	Новый	8-921-954-66-56
Кононенко Владимир Константинович	Новый	8-911-960-91-94
Гагарин Андрей Петрович	Постоянный	8-911-249-63-52
Балобеев Андрей Викторович	Новый	8-921-928-52-87
Бочаров Александр Вячеславович	Новый	8-921-900-32-41
Канзачаков Александр Владимирович	Постоянный	8-921-957-80-26
Салыкова Алия Кайратовна	Новый	8-911-794-30-72
Фрик Екатерина Олеговна	Новый	8-952-226-99-94
Качаев Эльгиз Идрисович	Новый	8-812-940-75-51
Кассюра Евгений Владимирович	Постоянный	8-911-995-84-81
Поддубная Светлана Николаевна	Новый	8-921-926-27-90
Наумова Анастасия Леонидовна	Постоянный	8-911-777-13-19
Дорофеева Ольга Андреевна	Новый	8-921-911-14-14
Котовская Анна Николаевна	Постоянный	8-911-923-06-83
Хотяков Виктор Владимирович	Постоянный	8-921-933-01-86
Рудюк Руслан Витальевич	Новый	8-921-959-40-40
Артемяева Елена Юрьевна	Новый	8-952-226-99-94
Белкин Вячеслав Михайлович	Новый	8-812-940-75-51

Рисунок 3.9 – Форма справочника «Клиенты»

Остальные справочники строятся по тому же принципу.

Теперь перейдем к операциям, доступным в системе (рисунок 3.10).

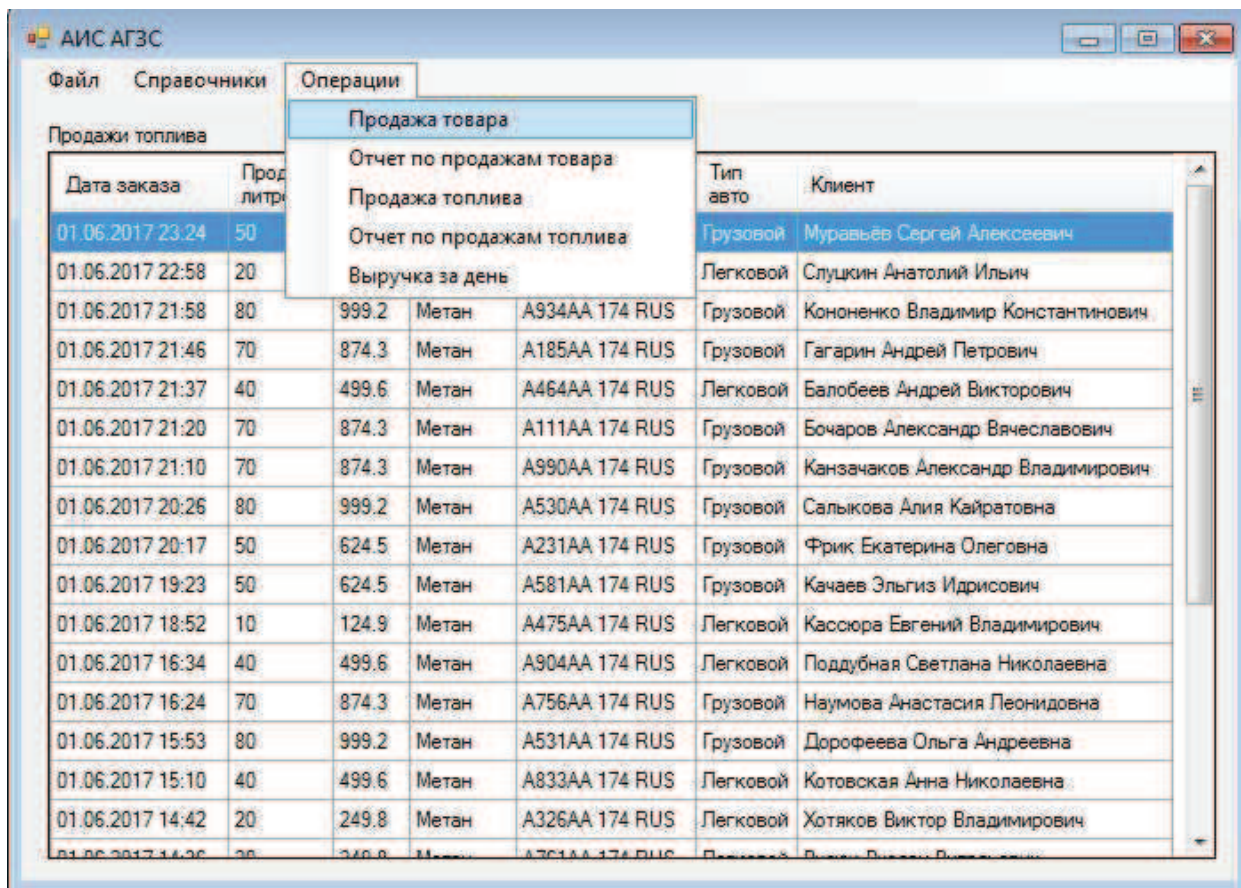


Рисунок 3.10 – Операции ИС

Как видно, данный список операций включает такие действия:

- внесение новой продажи товара в магазине;
- построение отчета по продажам товара;
- внесение новой продажи топлива;
- создание отчета по продажам топлива;
- получение выручки за день.

Покажем выполнение этих операций.

Итак, регистрация новой продажи товара в системе представлена на рисунке 3.11.

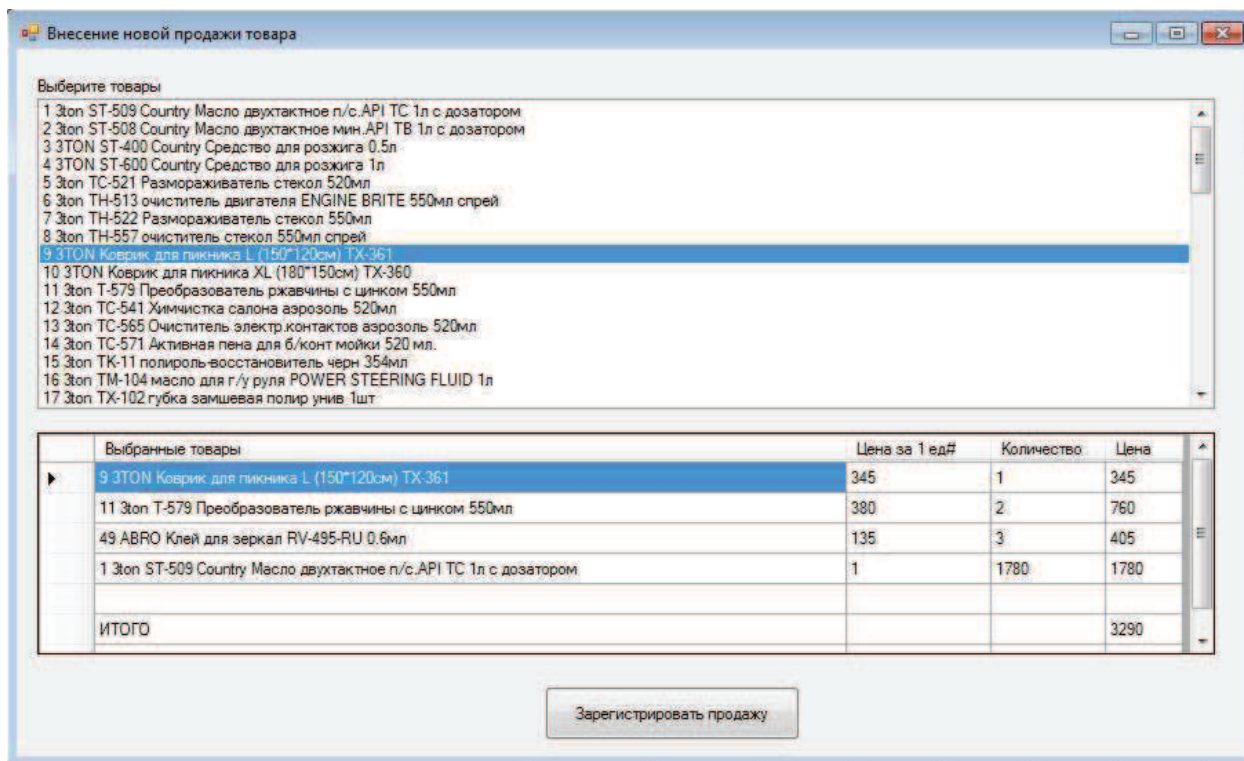


Рисунок 3.11 – Регистрация новой продажи товара в системе

После нажатия на кнопку «Зарегистрировать продажу» происходит уведомление оператора системы (рисунок 3.12).

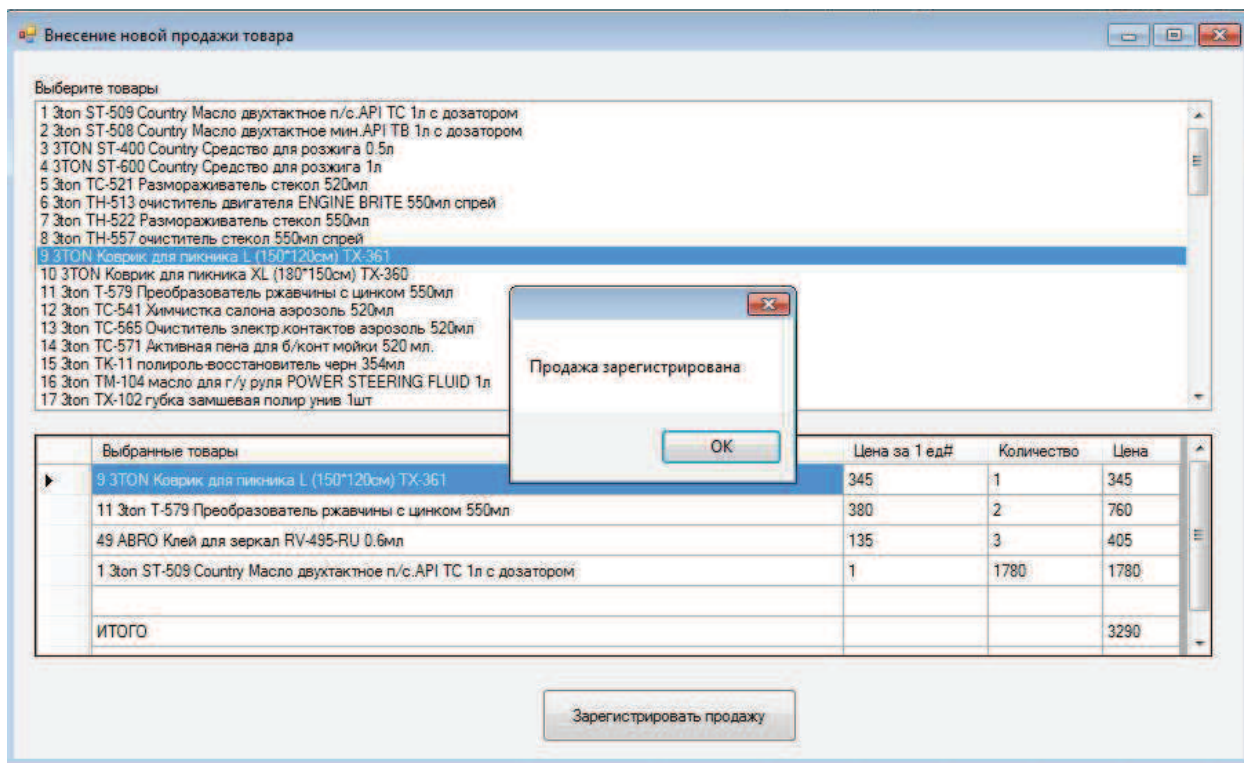


Рисунок 3.12 – Регистрация новой продажи товара прошла успешно

Изм.	Лист	№ докум.	Подпись	Дата
------	------	----------	---------	------

Также по мере накопления информации о продажах товара может потребоваться построить отчет по продажам за выбранный период (рисунок 3.13).

Дата	Сумма продаж по всем товарам, руб.
10.06.2017	12688
09.06.2017	22706
08.06.2017	17324
07.06.2017	11322
06.06.2017	15172
05.06.2017	15015
04.06.2017	17040
03.06.2017	21405
02.06.2017	12391
01.06.2017	15702
31.05.2017	25963
30.05.2017	21146
29.05.2017	25167
28.05.2017	10680
27.05.2017	14589
26.05.2017	18236
25.05.2017	28414

Рисунок 3.13 – Отчет по продажам товаров за период

Данный отчет можно выгрузить в Excel (рисунок 3.14).

Дата	Сумма продаж по всем товарам, руб.
10.06.2017	12688
09.06.2017	22706
08.06.2017	17324
07.06.2017	11322
06.06.2017	15172
05.06.2017	15015
04.06.2017	17040
03.06.2017	21405
02.06.2017	12391
01.06.2017	15702
31.05.2017	25963
30.05.2017	21146

Рисунок 3.14 – Отчет по продажам товаров в Excel

Продажа топлива клиенту осуществляется через специальную форму (рисунок 3.15). Нужно выбрать тип топлива, клиента, а затем ввести количество топлива. Система автоматически на основе данных о стоимости топлива посчитает сумму для оплаты клиента.

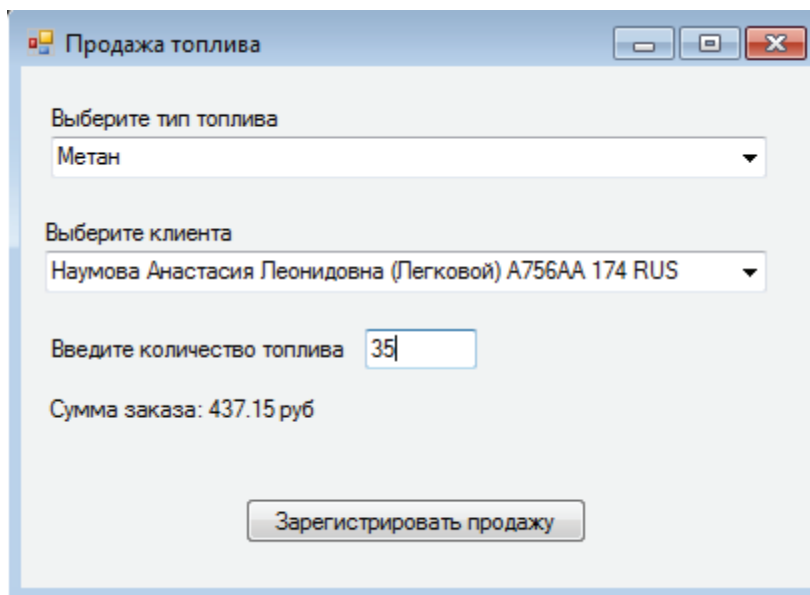


Рисунок 3.15 – Продажа топлива клиенту

После нажатия на кнопку «Зарегистрировать продажу» появляется сообщение оператору (рисунок 3.16).

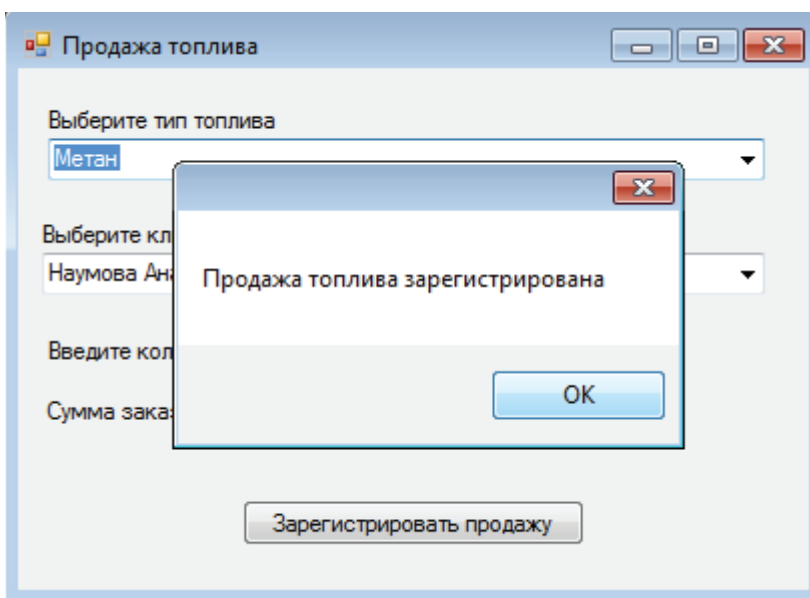
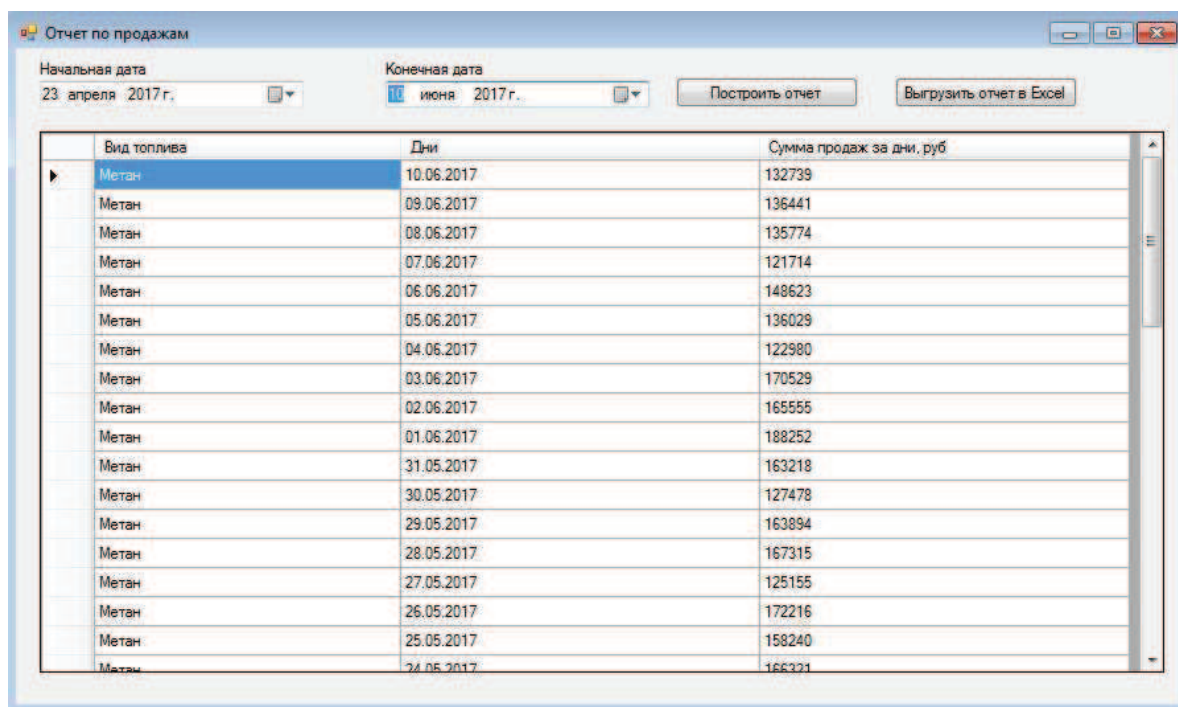


Рисунок 3.16 – Успешная продажа топлива

Система также позволяет построить отчет о продажах топлива за период (рисунок 3.17). В качестве периода выбран с 23.04.2017 по 10.06.2017. Система



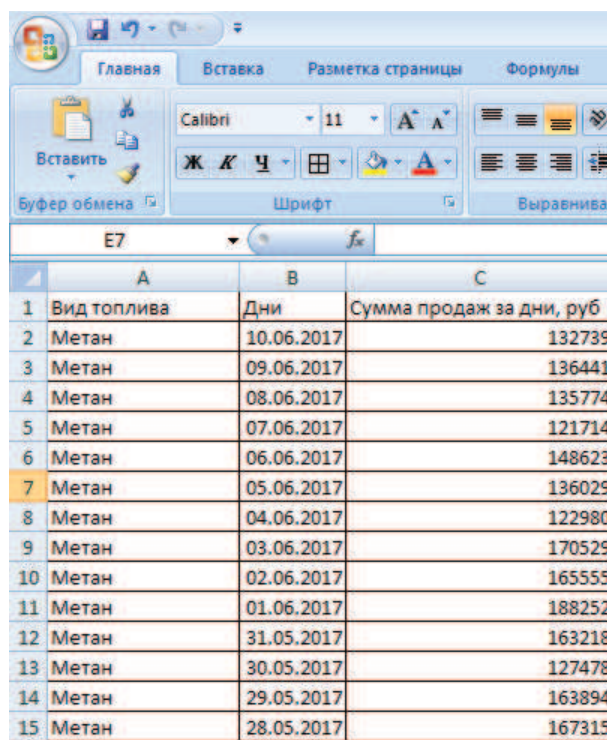
подсчитывает сумму продаж топлива за каждый день выбранного периода.



Вид топлива	Дни	Сумма продаж за дни, руб.
Метан	10.06.2017	132739
Метан	09.06.2017	136441
Метан	08.06.2017	135774
Метан	07.06.2017	121714
Метан	06.06.2017	148623
Метан	05.06.2017	136029
Метан	04.06.2017	122980
Метан	03.06.2017	170529
Метан	02.06.2017	165555
Метан	01.06.2017	188252
Метан	31.05.2017	163218
Метан	30.05.2017	127478
Метан	29.05.2017	163894
Метан	28.05.2017	167315
Метан	27.05.2017	125155
Метан	26.05.2017	172216
Метан	25.05.2017	158240
Метан	24.05.2017	166321

Рисунок 3.17 – Отчет по продажам топлива за период

Данный отчет можно выгрузить в Excel файл, что может быть полезно для бухгалтерии (рисунок 3.18).



	A	B	C
1	Вид топлива	Дни	Сумма продаж за дни, руб.
2	Метан	10.06.2017	132739
3	Метан	09.06.2017	136441
4	Метан	08.06.2017	135774
5	Метан	07.06.2017	121714
6	Метан	06.06.2017	148623
7	Метан	05.06.2017	136029
8	Метан	04.06.2017	122980
9	Метан	03.06.2017	170529
10	Метан	02.06.2017	165555
11	Метан	01.06.2017	188252
12	Метан	31.05.2017	163218
13	Метан	30.05.2017	127478
14	Метан	29.05.2017	163894
15	Метан	28.05.2017	167315

Рисунок 3.18 – Выгрузка отчета по продажам в Excel

Иногда руководству требуется быстро узнать сумму выручки на текущий

момент. Для этого есть специальная операция в меню операций: «Получение выручки за день».

При ее выполнении оператору системы выводится сообщение о текущей выручке (рисунок 3.19).

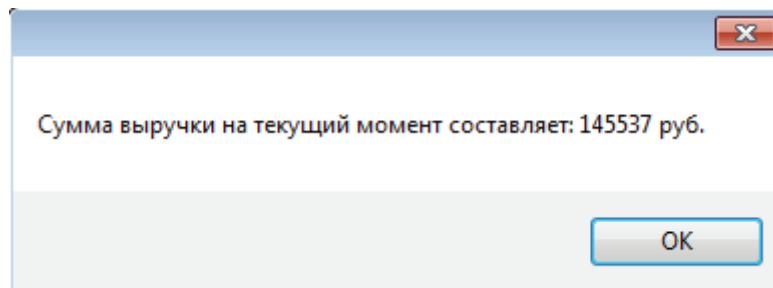


Рисунок 3.19 – Сообщение о текущей выручке за день

Таким образом, разработанная информационная система позволяет вести учет продаж топлива и товаров на АГЗС и строить отчеты для дальнейшего использования, например, бухгалтерией.

## ЗАКЛЮЧЕНИЕ

В рамках работы выполнено проектирование информационной системы, автоматизирующей деятельность АГЗС.

Объектом исследования является предприятие «АГЗС+». Предприятие в рамках своей деятельности реализует газовое топливо и бензин для автомобилей, а также продает товары из магазина на автозаправочных станциях.

Выполнен анализ организационной структуры предприятия, который показал, что:

В оргструктуре директору административно подчинено 4 человека. Менеджеру также подчинены 4 человека, поэтому они могут эффективно руководить.

Оргструктура является достаточно однородной. В целом же некоторая неоднородность организационной структуры допустима и даже неизбежна.

Оргструктура не имеет должностей, которые явно дублируют друг друга.

В организационной структуре присутствует некоторая избыточность руководящего персонала.

В рамках моделирования бизнес процессов построена модель в нотации IDEF0 в среде BPWin, а также в нотации DFD.

Исходя из проведенного моделирования бизнес процессов предприятия, касающихся продажи и учета топлива можно сделать вывод о том, что эти процессы выполняются недостаточно эффективно.

Для учета проданного топлива используется Excel файл, где каждая новая продажа представляет собой строку в файле.

Стоимость топлива рассчитывается вручную на калькуляторе, что приводит к периодическим ошибкам и недостаткам.

Поэтому предлагается внедрить на предприятии информационную систему, которая будет автоматизировать процесс учета и обработки данных по продажам топлива на АГЗС.

Составлено техническое задание на проект, где определены параметры

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		60

разработки и функции системы.

Рассмотренные и изученные аналоги проектируемой ИС обладают внушительным функционалом, выходящим за рамки требований рассматриваемого предприятия. К тому же программные продукты являются платными. С учетом того, что предприятие предъявляет не такие значительные требования к функционалу, самостоятельная разработка системы является обоснованной. По мере необходимости такая система в будущем может совершенствоваться путем добавления дополнительных модулей.

В качестве языка программирования выбран C#, которому соответствует среда разработки Visual Studio. Этот выбор обусловлен тем, что C# фактически интегрировал в себя все плюсы языков Java и C++, а также имеет такие дополнительные особенности, как автоматический сборщик мусора, продвинутую систему поддержки разработки кода, встроенную в IDE, множество удобных и полезных типов данных, значительно ускоряющих процесс разработки. К тому же IDE Visual C# Express является бесплатной средой разработки приложений на C#.

В качестве СУБД была выбрана MySQL, поскольку она самая распространенная полноценная серверная СУБД. Также MySQL очень функциональная, свободно распространяемая СУБД.

Составлена инфологическая и даталогическая модель БД, а также сформирован скрипт инициализации БД.

Макеты пользовательского интерфейса сделаны с помощью встроенного редактора в Microsoft Visual Studio.

Информационная система включает в себя 10 физических модулей. Представлена иерархическая схема модулей.

Тестирование приложения показало его работоспособность и соответствие требованиям, составленным в рамках постановки задачи на проектирование.

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
						61
Изм.	Лист	№ докум.	Подпись	Дата		

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Федеральный закон Российской Федерации от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации».
2. ГОСТ 34.602.89 Техническое задание на создание АС.
3. ГОСТ 34.601-90 АС. Стадии создания.
4. Вагнер Б. Эффективное программирование на С#. 50 способов улучшения кода. – М.: Вильямс, 2017. – 224 с.
5. Волкова В.Н. Теория систем и системный анализ. Учебное пособие. – М.: Юрайт, 2016. – 464 с.
6. Емельянова Н.З. Проектирование информационных систем. – М.: Форум, 2014. – 432 с.
7. Качала В.В. Теория систем и системный анализ. – М.: Academia, 2013. – 272 с.
8. Кориков А.М. Теория систем и системный анализ. Учебное пособие. – М.: НИЦ ИНФРА-М, 2016 – 288 с.
9. Культин Н.Б. Microsoft Visual С# в задачах и примерах. – М.: Петербург, 2015. – 320 с.
10. Маклафлин Б. PHP и MySQL. Исчерпывающее руководство. – М.: Питер, 2016. – 544 с.
11. Ригс С. Администрирование PostgreSQL 9. Книга рецептов. – М.: ДМК Пресс, 2015. – 364 с.
12. Стиллмен Э. Изучаем С#. – М.: Питер, 2016. – 816 с.
13. Форта Б. Oracle PL/SQL за 10 минут. – М.: Альфа-книга, 2016. – 336 с.
14. Tiffani R. SQL Server CE Database Development with the .NET Compact Framework, 2013. – 246 с.
15. Анализ организационной структуры предприятия – <http://www.fox-manager.ru/analiz-organizacionnoj-struktury-predpriyatiya.html>

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		62

16. Проектирование информационных систем –  
[http://www.lnau.lg.ua/dist/inf\\_12.html](http://www.lnau.lg.ua/dist/inf_12.html)
17. Проектирование и разработка автоматизированных, информационных и аналитических систем – [http://www.info-system.ru/designing/methodology/dfd/dfd\\_theory\\_dfd.html](http://www.info-system.ru/designing/methodology/dfd/dfd_theory_dfd.html)
18. Проектирование баз данных –  
[https://ru.wikipedia.org/wiki/Проектирование\\_баз\\_данных](https://ru.wikipedia.org/wiki/Проектирование_баз_данных)
19. Колесников С.Н. Как организовывать проект внедрения –  
<http://citforum.ru/cfin/articles/organize.shtml>
20. Колясников С.А. Корпоративные информационные системы. Взгляд изнутри – <http://uralsoft.by.ru/>
21. Информационная система – [https://ru.wikipedia.org/wiki/Информационная\\_система](https://ru.wikipedia.org/wiki/Информационная_система)
22. ERwin Data Modeler –  
[https://ru.wikipedia.org/wiki/ERwin\\_Data\\_Modeler](https://ru.wikipedia.org/wiki/ERwin_Data_Modeler)
23. Типы организационных структур управления –  
<http://pagelooker.org/rezume/2-uncategorised/16-tipy-organizatsionnykh-struktur-upravleniya-chast-1.html>

## ПРИЛОЖЕНИЕ А – СКРИПТ ИНИЦИАЛИЗАЦИИ БД

```
CREATE TABLE auto
(
    id_au  INTEGER NULL,
    name   VARCHAR(20) NULL
)
;
ALTER TABLE auto
    ADD PRIMARY KEY (id_au)
;
CREATE TABLE auto_types
(
    id_au_ty  INTEGER NULL,
    type      VARCHAR(20) NULL
)
;
ALTER TABLE auto_types
    ADD PRIMARY KEY (id_au_ty)
;
CREATE TABLE cards
(
    id_ca  INTEGER NULL,
    title  VARCHAR(20) NULL,
    date_created  VARCHAR(20) NULL,
    sum_orders  INTEGER NULL,
    id_di  INTEGER NULL,
    id_cl  INTEGER NULL,
    number  VARCHAR(20) NULL
)
;
ALTER TABLE cards
    ADD PRIMARY KEY (id_ca)
;
CREATE TABLE client_types
(
    id_cl_ty  INTEGER NULL,
    type      VARCHAR(20) NULL
)
;
ALTER TABLE client_types
    ADD PRIMARY KEY (id_cl_ty)
;
CREATE TABLE clients
(
    id_cl  INTEGER NULL,
    fio    VARCHAR(20) NULL,
    tel    VARCHAR(20) NULL,
    id_cl_ty  INTEGER NULL
)
;
ALTER TABLE clients
```

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		64

```

        ADD PRIMARY KEY (id_cl)
;
CREATE TABLE discounts
(
    id_di INTEGER NULL,
    discount INTEGER NULL
)
;
ALTER TABLE discounts
    ADD PRIMARY KEY (id_di)
;
CREATE TABLE earnings
(
    id_ea INTEGER NULL,
    earn_date VARCHAR(20) NULL,
    amount VARCHAR(20) NULL
)
;
ALTER TABLE earnings
    ADD PRIMARY KEY (id_ea)
;
CREATE TABLE fuel
(
    id_fu INTEGER NULL,
    name VARCHAR(20) NULL
)
;
ALTER TABLE fuel
    ADD PRIMARY KEY (id_fu)
;
CREATE TABLE fuel_income
(
    id_fu_in INTEGER NULL,
    date_time INTEGER NULL,
    cnt INTEGER NULL,
    id_fu INTEGER NULL,
    id_po INTEGER NULL
)
;
ALTER TABLE fuel_income
    ADD PRIMARY KEY (id_fu_in)
;
CREATE TABLE fuel_prices
(
    id_fu_pr INTEGER NULL,
    date_price VARCHAR(20) NULL,
    price INTEGER NULL,
    id_fu INTEGER NULL
)
;
ALTER TABLE fuel_prices
    ADD PRIMARY KEY (id_fu_pr)

```

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		65



```

;
CREATE TABLE fuel_storage
(
    id_fu_st  INTEGER NULL,
    cnt       VARCHAR(20) NULL,
    id_fu     INTEGER NULL
)
;
ALTER TABLE fuel_storage
    ADD PRIMARY KEY (id_fu_st)
;
CREATE TABLE indications
(
    id_in     INTEGER NULL,
    date_time VARCHAR(20) NULL,
    level_gauge VARCHAR(20) NULL,
    temperature VARCHAR(20) NULL
)
;
ALTER TABLE indications
    ADD PRIMARY KEY (id_in)
;
CREATE TABLE orders
(
    id_or     INTEGER NULL,
    datetime  DATE NULL,
    cnt_liters INTEGER NULL,
    price     VARCHAR(20) NULL,
    id_fu     INTEGER NULL,
    id_au     INTEGER NULL,
    id_cl     INTEGER NULL,
    id_st     INTEGER NULL,
    id_au_ty  INTEGER NULL
)
;
ALTER TABLE orders
    ADD PRIMARY KEY (id_or)
;
CREATE TABLE positions
(
    id_po     INTEGER NULL,
    position  CHAR(18) NULL
)
;
ALTER TABLE positions
    ADD PRIMARY KEY (id_po)
;
CREATE TABLE product_consumption
(
    id_pr_co  INTEGER NULL,
    dste_time VARCHAR(20) NULL,
    cnt       INTEGER NULL,

```

					<b>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		66

```

        id_pr  INTEGER NULL,
        name  CHAR(18) NULL
    )
;
ALTER TABLE product_consumption
    ADD PRIMARY KEY (id_pr_co)
;
CREATE TABLE product_income
(
    id_pr_in  INTEGER NULL,
    date_time VARCHAR(20) NULL,
    cnt       INTEGER NULL,
    id_pr     INTEGER NULL,
    name      CHAR(18) NULL
)
;
ALTER TABLE product_income
    ADD PRIMARY KEY (id_pr_in)
;
CREATE TABLE product_storage
(
    id_pr_st  INTEGER NULL,
    cnt       INTEGER NULL,
    id_pr     INTEGER NULL,
    name      CHAR(18) NULL
)
;
ALTER TABLE product_storage
    ADD PRIMARY KEY (id_pr_st)
;
CREATE TABLE products
(
    id_pr  INTEGER NULL,
    name  CHAR(18) NULL,
    info  INTEGER NULL,
    price INTEGER NULL
)
;
ALTER TABLE products
    ADD PRIMARY KEY (id_pr,name)
;
CREATE TABLE settings
(
    name  VARCHAR(20) NULL,
    value VARCHAR(20) NULL
)
;
ALTER TABLE settings
    ADD PRIMARY KEY (name)
;
CREATE TABLE stations
(

```

```

    id_st  INTEGER NULL,
    name   VARCHAR(20) NULL,
    address VARCHAR(20) NULL,
    tel    VARCHAR(20) NULL,
    info   VARCHAR(20) NULL,
    id_us  VARCHAR(20) NULL
)
;
ALTER TABLE stations
    ADD PRIMARY KEY (id_st)
;
CREATE TABLE users
(
    id_us  VARCHAR(20) NULL,
    fio    VARCHAR(20) NULL,
    login  VARCHAR(20) NULL,
    psswd  VARCHAR(20) NULL,
    id_po  INTEGER NULL
)
;
ALTER TABLE users
    ADD PRIMARY KEY (id_us)
;
ALTER TABLE cards
    ADD FOREIGN KEY R_9 (id_di) REFERENCES discounts(id_di)
;
ALTER TABLE cards
    ADD FOREIGN KEY R_10 (id_cl) REFERENCES clients(id_cl)
;
ALTER TABLE clients
    ADD FOREIGN KEY R_7 (id_cl_ty) REFERENCES
client_types(id_cl_ty)
;
ALTER TABLE fuel_income
    ADD FOREIGN KEY R_13 (id_fu) REFERENCES fuel(id_fu)
;
ALTER TABLE fuel_income
    ADD FOREIGN KEY R_14 (id_po) REFERENCES positions(id_po)
;
ALTER TABLE fuel_prices
    ADD FOREIGN KEY R_11 (id_fu) REFERENCES fuel(id_fu)
;
ALTER TABLE fuel_storage
    ADD FOREIGN KEY R_12 (id_fu) REFERENCES fuel(id_fu)
;
ALTER TABLE orders
    ADD FOREIGN KEY R_2 (id_fu) REFERENCES fuel(id_fu)
;
ALTER TABLE orders
    ADD FOREIGN KEY R_3 (id_au) REFERENCES auto(id_au)
;
ALTER TABLE orders

```

```

        ADD FOREIGN KEY R_4 (id_cl) REFERENCES clients(id_cl)
;
ALTER TABLE orders
        ADD FOREIGN KEY R_6 (id_st) REFERENCES stations(id_st)
;
ALTER TABLE orders
        ADD FOREIGN KEY R_8 (id_au_ty) REFERENCES auto_types(id_au_ty)
;
ALTER TABLE product_consumption
        ADD FOREIGN KEY R_17 (id_pr,name) REFERENCES
products(id_pr,name)
;
ALTER TABLE product_income
        ADD FOREIGN KEY R_16 (id_pr,name) REFERENCES
products(id_pr,name)
;
ALTER TABLE product_storage
        ADD FOREIGN KEY R_15 (id_pr,name) REFERENCES
products(id_pr,name)
;
ALTER TABLE stations
        ADD FOREIGN KEY R_5 (id_us) REFERENCES users(id_us)
;
ALTER TABLE users
        ADD FOREIGN KEY R_1 (id_po) REFERENCES positions(id_po)
;

```

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		69

ПРИЛОЖЕНИЕ Б – МАКЕТЫ ФОРМ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

Авторизация

IP адрес:порт сервера баз данных

Логин admin

Пароль \*\*\*\*\*

Войти

Рисунок Б.1 – Макет формы авторизации

Form1

Файл Справочники Операции

Продажи топлива

Рисунок Б.2 – Макет главной формы приложения

Изм.	Лист	№ докум.	Подпись	Дата

ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР

## ПРИЛОЖЕНИЕ В – ЛИСТИНГИ ПРОГРАММНЫХ МОДУЛЕЙ

### Листинг 1 – Функции работы с БД

```
public MySqlConnection getConnection() // Получить соединение с БД
{
    try
    {
//if (conn == null)
        {
            MySqlConnection cn = new
MySqlConnection();
cn.ConnectionString = connString;
cn.Open();
connected = true;
return cn;
        }
/*else if (conn.State.ToString() == "Closed")
cn.Open();
return conn;*/
    }
    catch(MySqlException ex)
    {
if(DEBUG)
        MessageBox.Show("Ошибка подключения к БД:\r\n"
+ ex.Message);
return conn;
    }
}

//
*****
*****

public bool closeConnection(MySqlConnection cn)
{
    try
    {
        if(!connected)
            return true;
cn.Close();
connected = false;
return true;
    }
    catch (Exception ex)
    {
        if (DEBUG)
            MessageBox.Show("Ошибка закрытия
соединения с БД:\r\n" + ex.Message);
return false;
    }
}
```

					ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР	Лист
Изм.	Лист	№ докум.	Подпись	Дата		71

```

//
*****
*****

        public bool MySqlSelect(String MySql, ref
List<List<String>> MySqlresult) // Выполнить запрос к БД
        {
            try
            {
                conn = getConnection();
                MySqlCommand myCommand = new
MySqlCommand(MySql, conn);
                MySqlDataReader rd = myCommand.ExecuteReader();
                MySqlresult.Clear();
                while(rd.Read())
                {
                    List<String> row = new List<String>();
                    for (int i = 0; i < rd.FieldCount; i++)
                    {
                        row.Add(rd.GetValue(i).ToString());
                    }
                MySqlresult.Add(row);
            }
                closeConnection(conn);
            }
            catch (Exception ex)
            {
                if (DEBUG)
                    MessageBox.Show("Ошибка выполнения MYSQL
запроса:\r\n"+MySql+"\r\n"+ex.Message);
                return false;
            }
            return true;
        }

//
*****
*****

        public int MySqlInsert(String MySql) // Выполнить запрос
к БД
        {
            try
            {
                conn = getConnection();
                MySqlDataAdapter da = new MySqlDataAdapter();
                da.InsertCommand = new MySqlCommand(MySql);
                da.InsertCommand.Connection = conn;
                da.InsertCommand.ExecuteNonQuery();
                da.SelectCommand = new MySqlCommand("SELECT
SCOPE_IDENTITY()", da.InsertCommand.Connection);
                int intID =

```

```

Convert.ToInt32(da.SelectCommand.ExecuteScalar());
        closeConnection(conn);
        return intID;
    }
catch (Exception ex)
    {
        if (DEBUG)
            MessageBox.Show("Ошибка выполнения MYSQL
запроса:\r\n"+MySql+"\r\n"+ex.Message);
        return -1;
    }
}

//
*****
*****

public bool MySqlOther(String MySql) // Выполнить запрос
к БД
    {
        try
        {
            conn = getConnection();
            MySqlDataAdapter da = new MySqlDataAdapter();
            da.InsertCommand = new MySqlCommand(MySql);
            da.InsertCommand.Connection = conn;
            da.InsertCommand.ExecuteNonQuery();
            return true;
        }
catch (Exception ex)
    {
        if (DEBUG)
            MessageBox.Show("Ошибка выполнения MYSQL запроса:\r\n" + MySql +
"\r\n" + ex.Message);
        return false;
    }
}

```



## Листинг 2 – Класс выгрузки данных в Excel

```
class Excel
{
private string excelFilePath = string.Empty;
private int rowNumber = 1;

Microsoft.Office.Interop.Excel.Application myExcelApplication;
Microsoft.Office.Interop.Excel.Workbook myExcelWorkbook;
Microsoft.Office.Interop.Excel.Worksheet myExcelWorkSheet;

public string ExcelFilePath
{
get { return excelFilePath; }
set { excelFilePath = value; }
}

public int Rownumber
{
get { return rowNumber; }
set { rowNumber = value; }
}

public void openExcel(string path)
{
excelFilePath = path;

myExcelApplication = null;

myExcelApplication = new
Microsoft.Office.Interop.Excel.Application();
myExcelApplication.DisplayAlerts = false;

myExcelWorkbook =
(Microsoft.Office.Interop.Excel.Workbook) (myExcelApplication.Workb
ooks._Open(excelFilePath, System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value));

int numberOfWorkbooks = myExcelApplication.Workbooks.Count;

myExcelWorkSheet =
(Microsoft.Office.Interop.Excel.Worksheet)myExcelWorkbook.Workshee
ts[1];

int numberOfSheets = myExcelWorkbook.Worksheets.Count;
```

					<b>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		74

```

}

public void addDataToExcel(int i, object j, string value)
{

myExcelWorkSheet.Cells[i, j] = value;
}

public void closeExcel(string path)
{
try
{
myExcelWorkbook.SaveAs(excelFilePath,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
Microsoft.Office.Interop.Excel.XlSaveAsAccessMode.xlNoChange,
System.Reflection.Missing.Value, System.Reflection.Missing.Value,
System.Reflection.Missing.Value,
System.Reflection.Missing.Value, System.Reflection.Missing.Value);

myExcelWorkbook.Close(true, path,
System.Reflection.Missing.Value);

}
finally
{
if (myExcelApplication != null)
{
myExcelApplication.Quit();
}
}

}
}

```

					<i>ЮУрГУ-09.03.01.62.2017.265 ПЗ ВКР</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>		75

# ПРИЛОЖЕНИЕ Г – ДИАГРАММА КЛАССОВ

