

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего
образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Бизнес информатика

(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)

на тему Разработка информационной системы управления заказами клиентов
фирмы по доставке продуктов (на примере ООО «Элит Эскорт»)

Студент

А.Г. Деревянко

(И.О. Фамилия)

(личная подпись)

Руководитель

В.Ф. Глазова

(ученая степень, звание, И.О. Фамилия)

Тольятти 2021

Аннотация

Выпускная квалификационная работа по теме «Разработка информационной системы управления заказами фирмы по доставке продуктов (на примере ООО «Элит Эскорт»)» содержит 82 страницы текстового документа, 1 приложение, 44 рисунка, 12 таблиц, 22 использованных источника.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка литературы и приложений.

Актуальность исследования выпускной квалификационной работы заключается в том, что автоматизация бизнес-процессов сегодня очень важна для предприятий любой сферы деятельности и любого масштаба, и эффективное решение в части управления заказами поможет сделать деятельность компании более привлекательной для клиентов и при этом более прибыльной.

Целью выпускной квалификационной работы является разработка информационной системы управления заказами клиентов фирмы по доставке продуктов на примере ООО «Элит Эскорт».

Задачи выпускной квалификационной работы заключаются в рассмотрении деятельности ООО «Элит Эскорт» по управлению заказами и автоматизации этой деятельности.

Результатом проектирования является автоматизированная информационная система управления заказами для ООО «Элит Эскорт», разработанная на языке программирования Delphi. База данных реализована в среде СУБД MS Access.

Оглавление

Введение.....	4
Глава 1 Функциональное моделирование предметной области	6
1.1 Техничко-экономическая характеристика предметной области.....	6
1.2 Концептуальное моделирование предметной области	9
1.3 Анализ существующих разработок на предмет соответствия сформулированным требованиям.....	21
1.4 Постановка задачи на разработку проекта создания информационной системы.....	25
1.5 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»	26
Глава 2 Логическое проектирование информационной системы	31
2.1 Выбор технологии логического моделирования	31
2.2 Логическая модель АИС и ее описание.....	32
2.3 Информационное обеспечение АИС.....	34
2.4 Проектирование базы данных информационной системы	41
2.5 Требования к аппаратно-программному обеспечению.....	45
Глава 3 Физическое проектирование информационной системы	48
3.1 Выбор архитектуры информационной системы	48
3.2 Выбор технологии разработки программного обеспечения информационной системы	49
3.3 Выбор СУБД.....	50
3.4 Разработка физической модели данных	52
3.5 Разработка программного обеспечения.....	56
3.6 Описание функциональности информационной системы.....	60
3.7 Тестирование программного продукта	63
Заключение	66
Список используемой литературы и используемых источников	67
Приложение А Код программы	69

Введение

Современный человек привык во многом полагаться на окружающую его микропроцессорную технику от мобильного телефона и фитнес-браслета до персонального компьютера и 3d-принтера. Автоматизированные информационные системы помогают быстро найти необходимую информацию, выбрать и заказать услуги и товары, провести различные платежи. Процесс взаимодействия с информационной системой удобен и лишен эмоционального фона взаимодействия между людьми, иногда несущего отрицательный оттенок.

Развитие аппаратного и программного обеспечения современных компьютеров, распространение компьютерных сетей, беспроводных видов связи, появление возможности пользоваться услугами глобальной сети Интернет – все это позволяет выполнять многие задачи, связанные с обработкой информации, практически из любого места, получая значительную экономию драгоценного времени.

Совершенствование информационных процессов – одна из важнейших задач любого современного предприятия, от самых маленьких компаний до промышленных гигантов. Развитие ИТ-структуры организации, автоматизация основных и обслуживающих бизнес-процессов влияют сегодня на общую успешность компании и ее эффективность, то есть фактически определяют наряду с базовыми производственными параметрами, приносит ли она прибыль.

Актуальность исследования выпускной квалификационной работы заключается в том, что автоматизация бизнес-процессов сегодня очень важна для предприятий любой сферы деятельности и любого масштаба, и эффективное решение в части управления заказами поможет сделать деятельность компании более привлекательной для клиентов и при этом более прибыльной.

Объектом исследования выпускной квалификационной работы являются бизнес-процессы компании по доставке продуктов ООО «Элит Эскорт».

Предметом исследования выпускной квалификационной работы выступает автоматизация управления заказами компании.

Целью выпускной квалификационной работы является разработка информационной системы управления заказами клиентов фирмы по доставке продуктов на примере ООО «Элит Эскорт».

Задачи выпускной квалификационной работы:

- провести анализ бизнес-процессов ООО «Элит Эскорт» в части управления заказами;
- разработать концепцию автоматизации процесса управления заказами;
- разработать автоматизированную информационную систему управления заказами для ООО «Элит Эскорт»;
- выполнить отладку и тестирование разработанной системы.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка литературы и приложений.

Первая глава работы отражает состояние бизнес-процессов исследуемой организации в части управления заказами потребителей и представляет модель реинжиниринга этого процесса.

Во второй главе ВКР представлено логическое проектирование автоматизированной информационной системы, сформулированы требования к разрабатываемой системе.

Третья глава посвящена разработке и тестированию автоматизированной системы управления заказами ООО «Элит Эскорт».

В заключении представлены главные из полученных по итогам выполнения ВКР результаты и практическая значимость разработки.

Глава 1 Функциональное моделирование предметной области

1.1 Техничко-экономическая характеристика предметной области

Общество с ограниченной ответственностью – это учрежденное одним или несколькими юридическими и/или физическими лицами хозяйственное общество, уставной капитал которого разделен на доли. При этом участники общества – учредители – не отвечают по его обязательствам и несут риск убытков, связанных с деятельностью общества, в пределах стоимости принадлежащих им долей или акций в уставном капитале общества.

Общество с ограниченной ответственностью «Элит Эскорт» предоставляет населению услуги, связанные с деятельностью в части организации общественного питания.

Общество с ограниченной ответственностью «Элит Эскорт» (ООО «Элит Эскорт») зарегистрировано в качестве юридического лица 01 сентября 2008 года. Организация имеет единственного учредителя, который с момента возникновения является ее руководителем – Деревянко О.С.

Основной вид деятельности – 56.29: Деятельность предприятий общественного питания по прочим видам организации питания.

Компания также реализует дополнительные виды деятельности, такие, как оптовая и розничная торговля пищевыми продуктами, напитками и табачными изделиями, деятельность по доставке продуктов.

ООО «Элит Эскорт» ведет экономическую деятельность с переменным успехом. В 2020 году предприятие понесло убытки в связи с возникновением пандемии. До 2020 года доходы компании постепенно росли (рисунок 1).

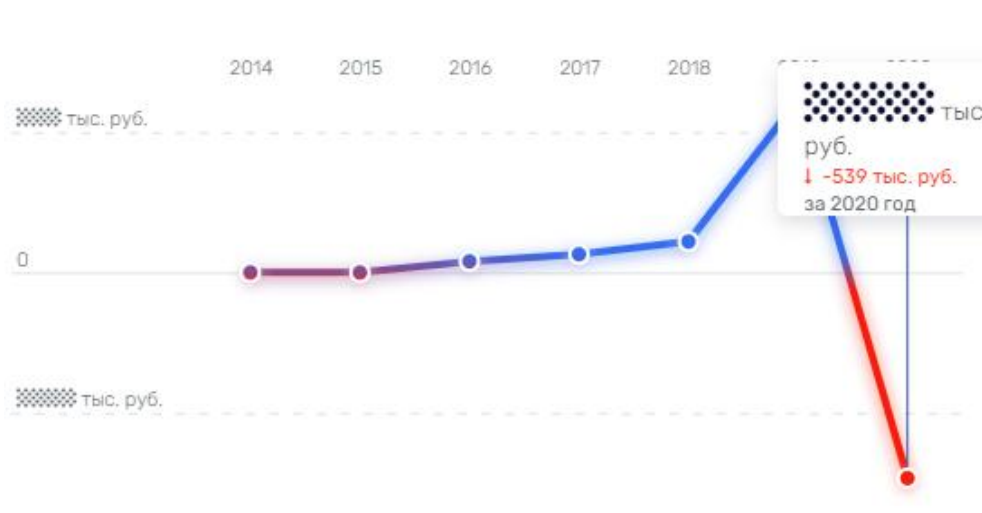


Рисунок 1 – Прибыль компании с 2014 года

В настоящее время ООО «Элит Эскорт» доставляет продукты питания и готовые блюда по заказу клиентов – преимущественно частных лиц, проживающих в рамках одного микрорайона.

В ООО «Элит Эскорт» работают на постоянной основе 12 сотрудников. В случае возникновения производственной необходимости к работе привлекаются дополнительные сотрудники на основании трудовых договоров.

Клиентами компании являются преимущественно частные лица, проживающие в шаговой доступности от офиса компании.

Компания снимает два смежных помещения в бизнес-центре. Одно помещение – офисное, в нем расположены рабочие места директора и диспетчеров. Второе помещение – подсобное, в нем продуктовые пакеты комплектуются на основе заказа. В этом же помещении созданы комфортные условия ожидания заказов для курьеров, упаковщиков и закупщиков.

Организационная структура ООО «Элит Эскорт» представлена на рисунке 2.

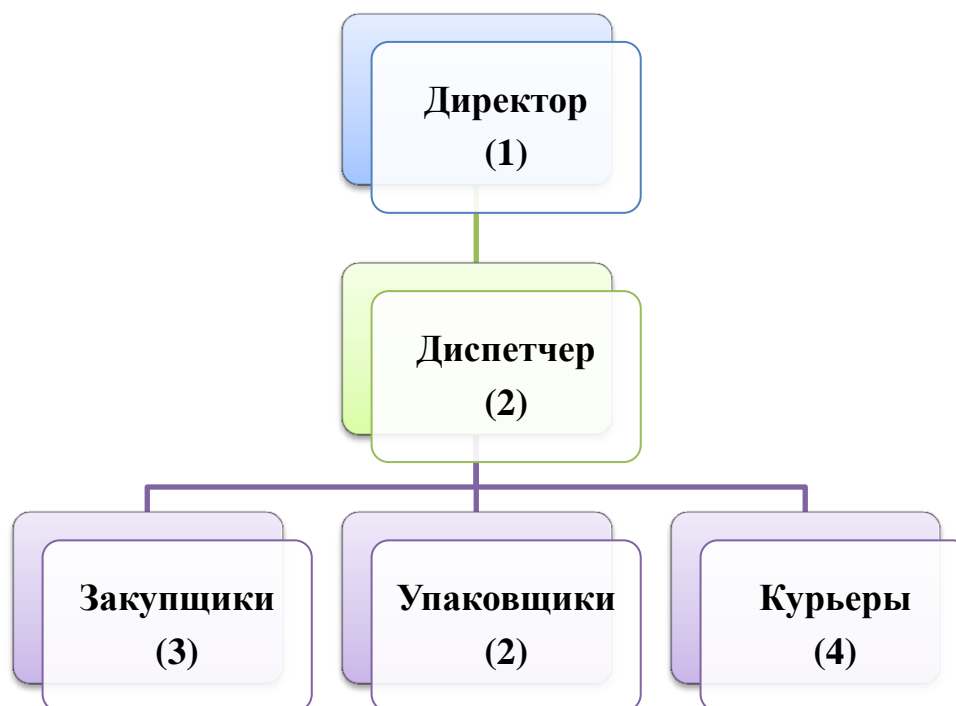


Рисунок 2 – Организационная структура ООО «Элит Эскорт»

Функции бухгалтера выполняет директор (он имеет соответствующее образование).

Техническое обслуживание офиса (уборка помещения, текущий ремонт, выполнение мелких сервисных работ) осуществляется соответствующей службой бизнес-центра, в котором арендуется офисное помещение для компании, и входит в арендную плату.

Директор осуществляет общее руководство компанией, определяет цели и задачи деятельности, разрабатывает стратегию его развития, ставит тактические задачи, совместно с диспетчерами анализирует эффективность работы организации и намечает пути ее повышения.

Диспетчер принимает заявки от клиентов, распределяет работы между сотрудниками, организует оптимальный режим взаимодействия между поставщиками продукции, сотрудниками компании и потребителями. Ведет учет реализованных заказов, а также работ, выполненных сотрудниками предприятия. Принимает обращения клиентов, отвечает на них в рамках своих компетенций или передает директору.

Закупщики осуществляют закупку продуктов у поставщиков на основании списка продуктов, полученного от диспетчера, и доставляют товар в офис.

Упаковщики согласно заказу формируют необходимые наборы продуктов, упаковывают их в фирменную тару компании и передают курьерам.

Курьеры доставляют сформированные заказы потребителям, получают оплату и обратную связь от них.

1.2 Концептуальное моделирование предметной области

1.2.1 Выбор технологии концептуального моделирования предметной области

Методы моделирования играют значительную роль в обеспечении поступательного развития организаций. Одним из методов моделирования является концептуальное моделирование. Применение моделей позволяет решить ряд актуальных практических задач как управленческого характера, так и оценочного и прогнозного.

«Моделированием называют создание объекта, схожего с изучаемым в определенных, выбранных исследователем, качествах, важных в рамках конкретного изыска. В разных исследованиях один и тот же моделируемый объект может быть представлен разными моделями.

В общем случае моделью считается материальный или нематериальный, мысленно воображаемый объект, замещающий в ходе изучения объект-оригинал, при условии фиксации отдельных, важных в контексте исследования, его качеств и характеристик.

Концептуальной моделью принято называть содержательную модель, при формулировке которой используются понятия и представления предметных областей знания, занимающихся изучением объекта моделирования» [7].

Специалисты сегодня выделяют следующие технологии и методологии концептуального моделирования информационных систем:

- структурный подход;
- объектно-ориентированный подход.

Основа структурного подхода опирается на декомпозицию (разбиение): система разбивается на подсистемы, которые также делятся на более мелкие элементы. Декомпозиция продолжается до конкретных процедур, при этом описываемая схема сохраняет целостное представление.

Наиболее востребованные модели:

- SADT (Structured Analysis and Design Technique) модели и соответствующие функциональные диаграммы;
- DFD (Data Flow Diagrams) – диаграммы потоков данных;
- ERD (Entity-Relationship Diagrams) диаграммы «сущность-связь».

Совокупность представленных технологий охватывают полное определение информационной системы.

Объектно-ориентированный подход представлен объектно-ориентированным анализом и проектированием (ООАП), который строится на объектно-ориентированной методологии представления предметной области в форме объектов, являющихся экземплярами различных классов.

Популярные графические нотации, использующиеся при объектно-ориентированном подходе – ERD и SADT.

В ходе анализа предметной области был выбран подход, позволяющий наиболее полно отразить особенности проектируемой системы. Предпочтение отдано методологии структурного анализа и проектирования информационных систем.

Для проектирования информационной системы выбран универсальный векторный графический редактор MS Visio и графические нотации DFD и IDEF0.

Диаграммы DFD (диаграмма потоков данных) описывает внешние по отношению к системе источники. Диаграммы IDEF0 используют для

описания бизнес-процессов и описания информации об использовании ресурсов.

Схемы IDEF0 обычно включают следующие компоненты:

- контекстная диаграмма — самая верхняя схема в модели IDEF0;
- иерархия декомпозиции IDEF0 с использованием связей между узлами-предками и узлами-потомками.

1.2.2 Моделирование бизнес-процессов предметной области для постановки задачи автоматизированного варианта решения

Основной бизнес-процесс, реализуемый ООО «Элит Эскорт» – доставка продуктов под заказ.

Для понимания организации бизнес-процесса разработана контекстная диаграмма деятельности AS-IS (модель КАК ЕСТЬ) и диаграммы, декомпозирующие бизнес-процесс до двух уровней.

Контекстная диаграмма деятельности AS-IS представлена на рисунке 3.

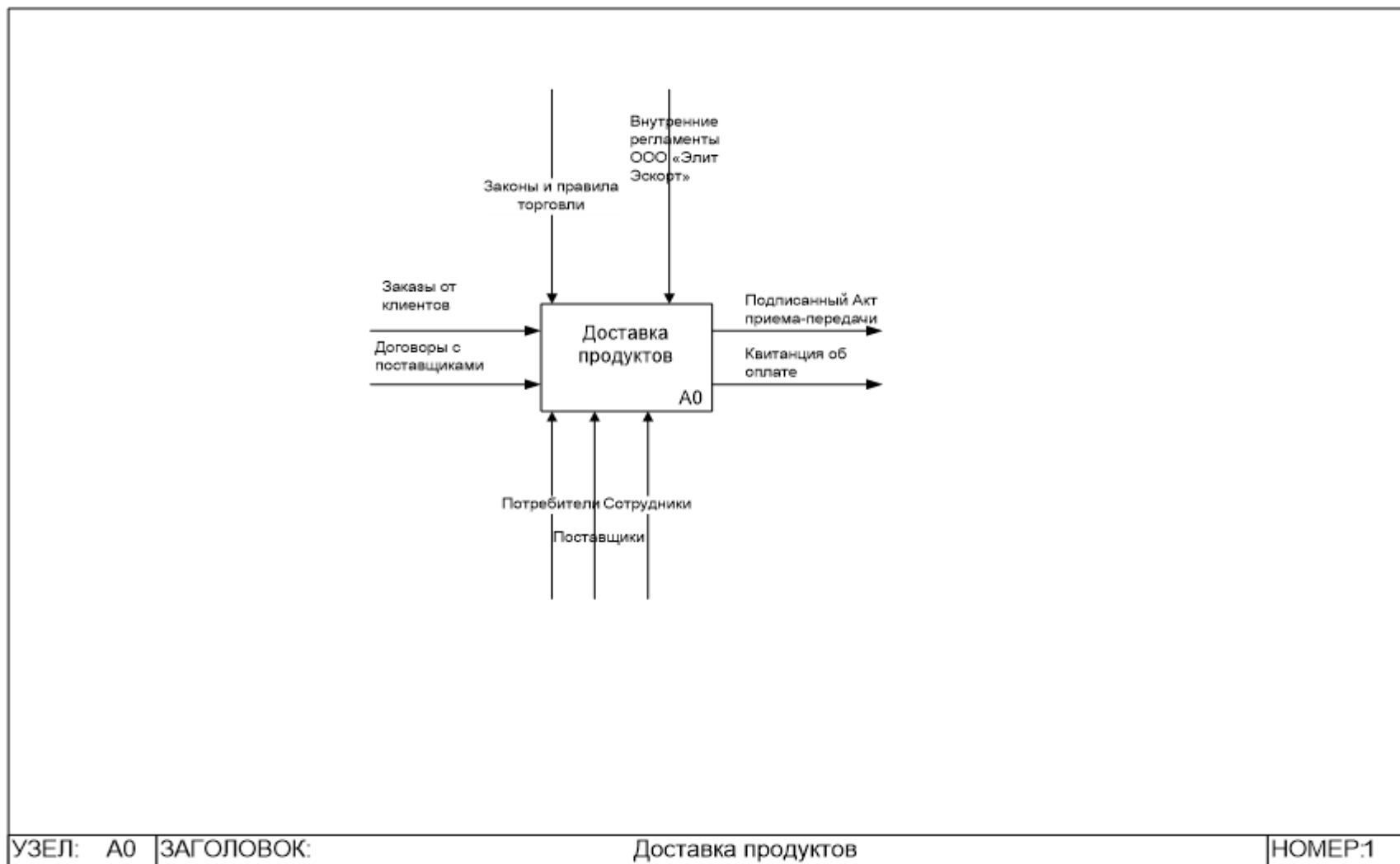


Рисунок 3 – Контекстная диаграмма бизнес-процесса доставки продуктов под заказ

В реализации основного бизнес-процесса можно выделить три этапа:

- получение и обработка заявки;
- формирование заказа;
- доставка заказа клиенту.

В процессе получения и обработки заявки диспетчер принимает заявки от клиентов по телефону. Собственного интернет-ресурса и/ или мобильного приложения ООО «Элит Эскорт» не имеет, так как в настоящее время в этом нет необходимости – основной контингент заказчиков составляют жители микрорайона среднего и пожилого возраста, которым комфортнее сделать заказ по телефону. Компании доверяют, поэтому имеется круг постоянных клиентов, которые делают заказы в течение нескольких лет. Это позволяет компании достаточно эффективно планировать закупки. С другой стороны, целью организации является увеличение прибыли, что может быть достигнуто только путем увеличения количества заявок, то есть требуется расширение контингента и, возможно, географии заказчиков.

Формирование заказа происходит в два этапа – закупка нужных продуктов (продукты заказываются поставщикам заранее с учетом статистики по сложившимся объемам продаж) и упаковка готовых заказов. Первый этап выполняют закупщики, второй – упаковщики.

После того, как заказ сформирован и упакован, он передается курьеру для доставки клиенту. Так как расстояния доставки в настоящий момент небольшие, не превышают 12 км, то курьеры доставляют продуктовые наборы в пешем порядке или на велосипедах (самокатах и т.п.). Доставка продуктов на автомобилях в настоящее время не практикуется.

Курьер, доставив заказ клиенту, получает оплату, получает подпись клиента о доставке заказа. Также в обязанности курьера включается получение отзыва от клиента о качестве продуктов и доставки.

Декомпозиции первого и второго уровней бизнес-процесса доставки продуктов представлены на рисунках 4 – 7.

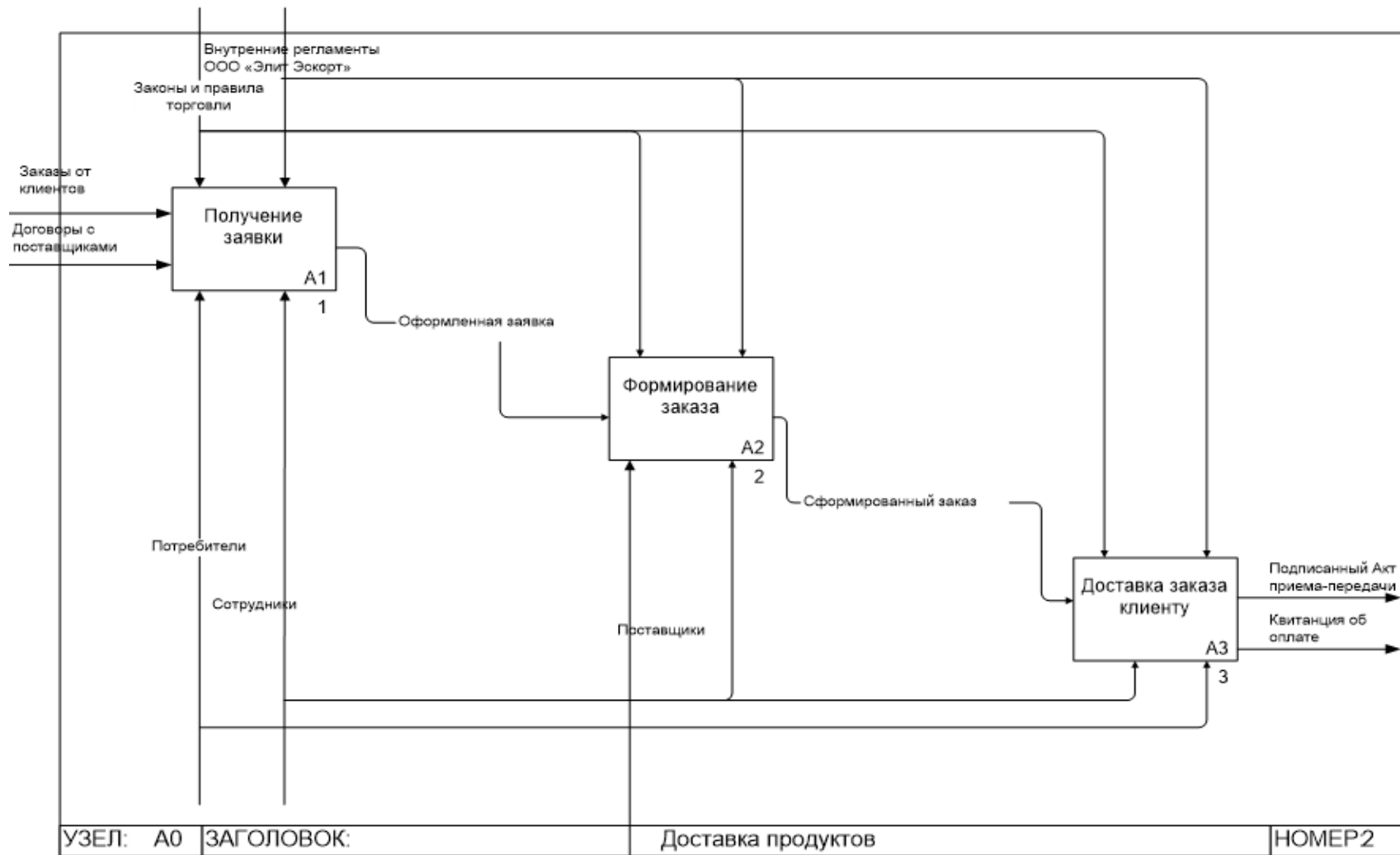


Рисунок 4 – Декомпозиция основного бизнес-процесса первого уровня

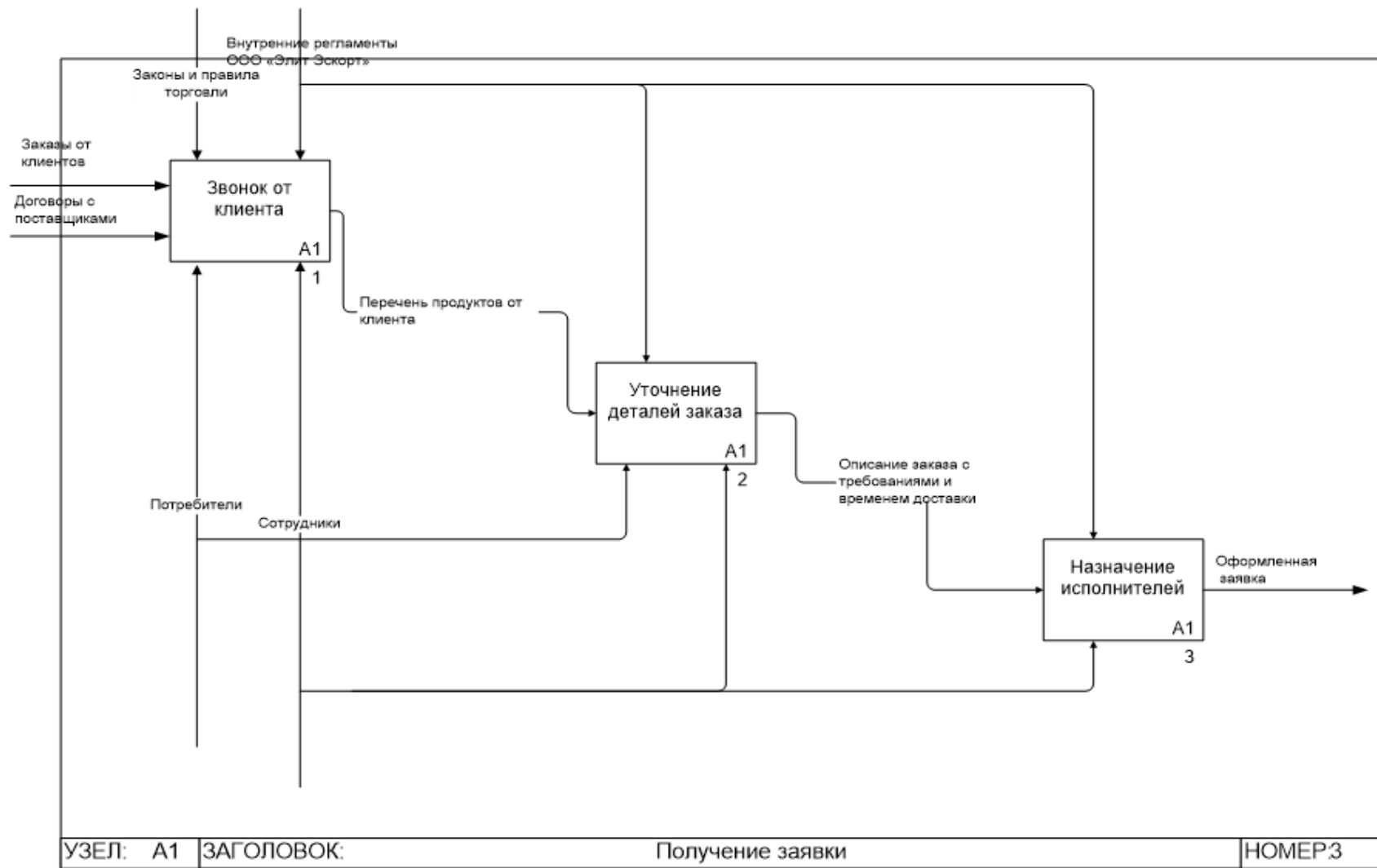


Рисунок 5 – Декомпозиция бизнес-процесса «Получение заявки»

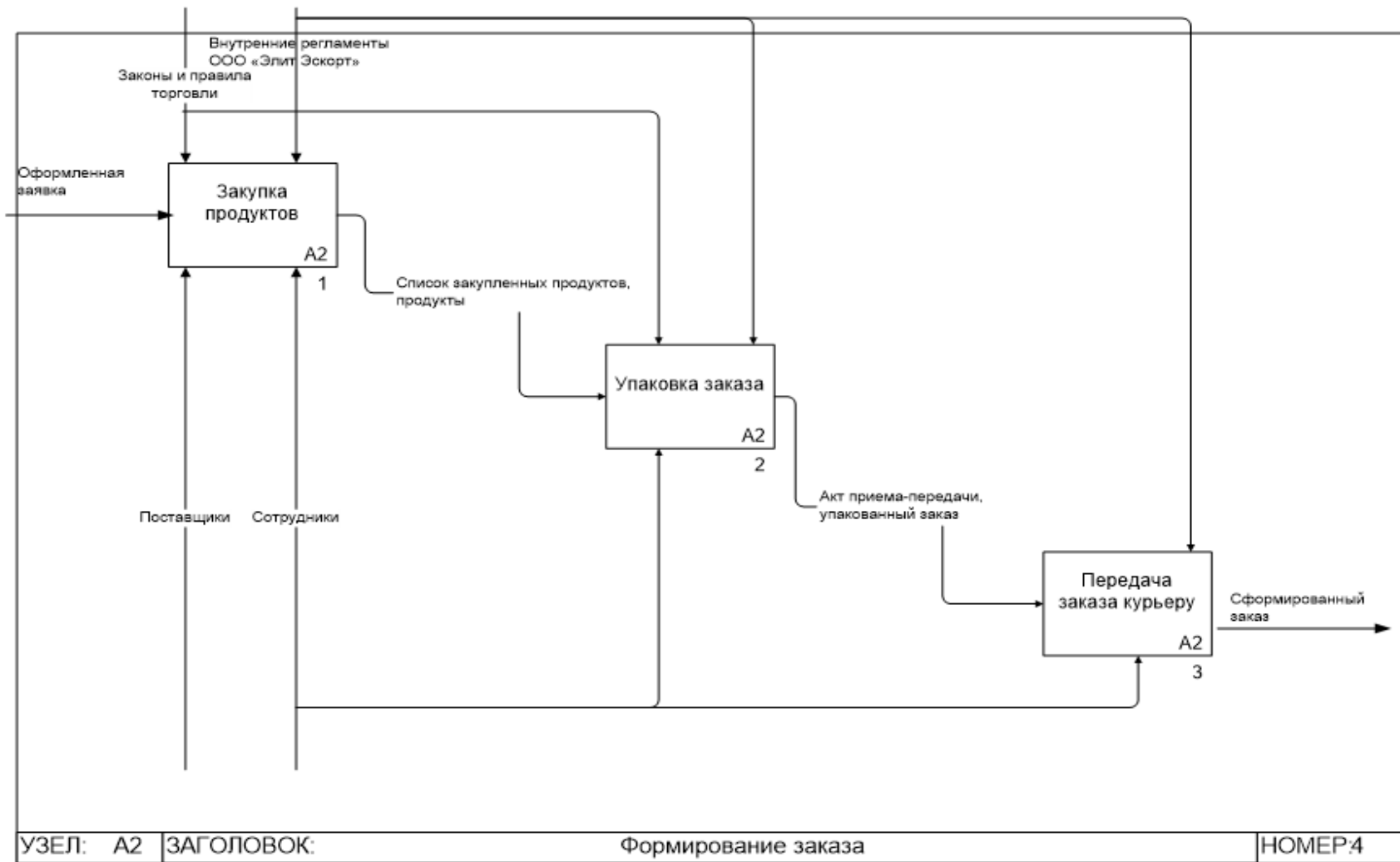


Рисунок 6 – Декомпозиция бизнес-процесса «Формирование заказа»

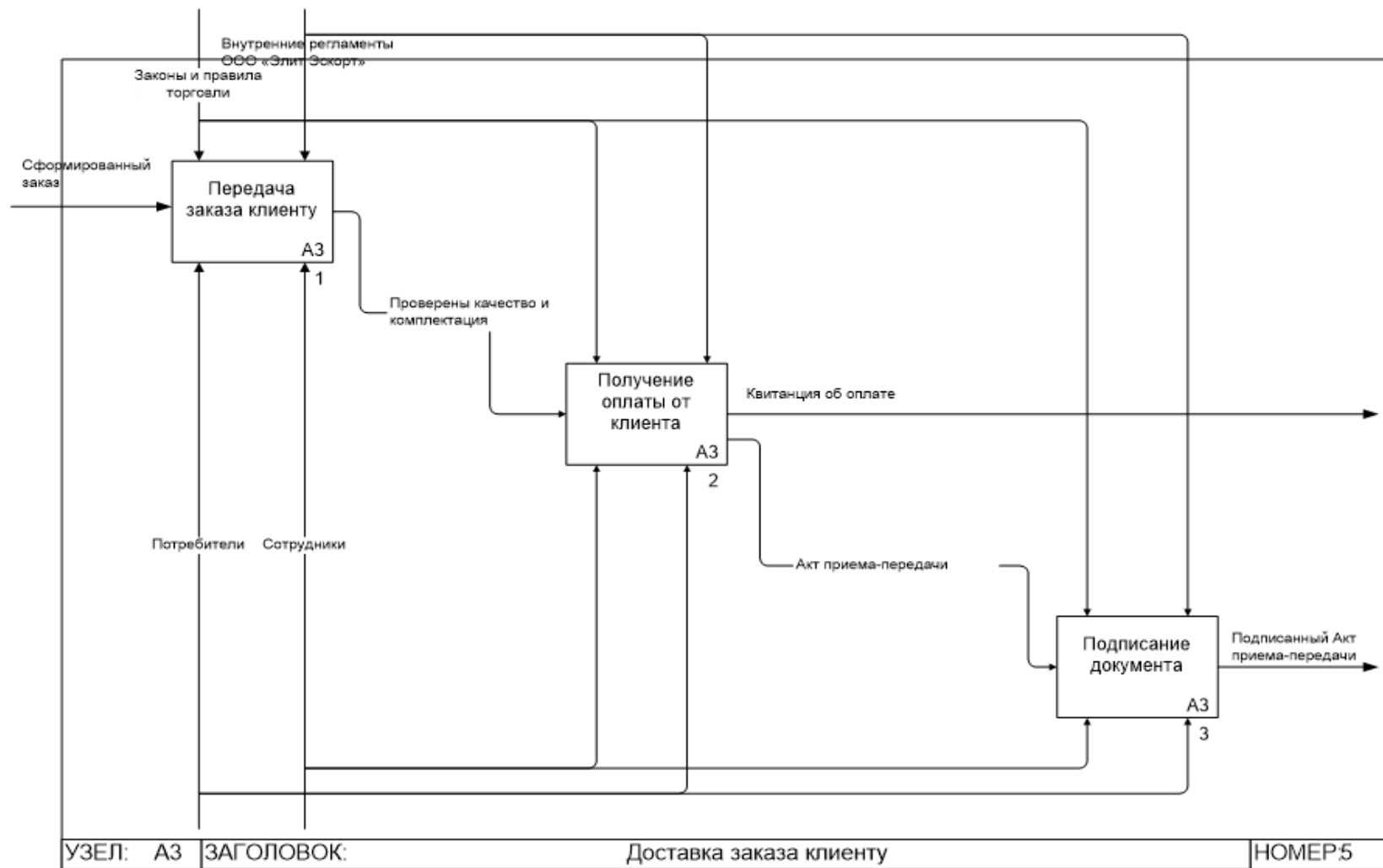


Рисунок 7 – Декомпозиция бизнес-процесса «Доставка заказа клиенту»

Разработанные диаграммы наглядно представляют бизнес-процесс доставки продуктов, являющийся основным в компании ООО «Элит Эскорт». Анализ модели бизнес-процесса доставки продуктов КАК ЕСТЬ позволит выявить уязвимые места и предложить возможные направления совершенствования процесса.

1.2.3 Анализ модели бизнес-процесса «КАК ЕСТЬ»

Входом процесса доставки продуктов является звонок клиента для заказа услуги. Звонок принимает диспетчер, он записывает информацию по звонку в произвольной форме (это может быть как бумажный формат, так и записи в таблице Excel) – установленных требований организации к ней нет.

В настоящее время, когда заказов не очень много и они имеют равномерно распределенный во времени характер, диспетчер успевает принять все заказы.

Но при этом требуется систематизировать информацию о заказах, что позволит работать более рационально и быстрее обслуживать клиентов. Если поток заказов увеличится, что является целью развития предприятия, то будет необходимо заключить договор с колл-центром о предоставлении услуги приема заказов.

После того, как диспетчер принял заказ и уточнил с клиентом его детали, он назначает исполнителей заказа – закупщика, упаковщика и курьера.

При этом диспетчер обеспечивает оптимальную логистику закупок и доставки – при наличии возможности совмещаются закупки и доставка в рамках разных заказов с аналогичным набором продуктов.

Действующий график соотношения времени заказа и времени доставки представлен в таблице 1.

Таблица 1 – Соотношение времени заказа и времени доставки

№	Время заказа	Время доставки
1.	9.00 – 10.00	12.00 – 13.00
2.	10.00 – 11.00	13.00 – 14.00
3.	11.00 – 12.00	14.00 – 15.00
4.	12.00 – 13.00	15 – 16.00
5.	13.00 – 14.00	16.00 – 17.00
6.	После 14.00	10.00 – 12.00 на следующий день

Назначения диспетчер передает сотрудникам в форме устного сообщения, если будущий исполнитель находится в офисе, или в форме записи на маркерной доске общего пользования. Иногда диспетчер звонит сотруднику и сообщает о назначении на заказ по телефону. Такой подход работает в условиях ограниченного потока заявок. Но при этом все же случается путаница, заказы теряются, случаются недоразумения между сотрудниками. Такие ситуации требуется исключить, а исключить их можно только систематизацией работы с заказом от его получения до полного исполнения.

Опираясь на модель бизнес-процесса КАК ЕСТЬ и ее анализ, можно сформировать требования к автоматизированной информационной системе для ООО «Элит Эскорт».

1.2.4 Обоснование необходимости автоматизированного варианта решения и формирование требований к новой технологии

Бизнес-процесс доставки продуктов в ООО «Элит Эскорт» в настоящее время работает, хотя и со сбоями только потому, что количество потребителей ограничивается небольшим кругом лиц. При увеличении количества клиентов без жесткого регламента каждой операции и без использования вычислительной техники при действующем порядке возникнет путаница в заказах, их составе и заказчиках. Сотрудники также могут испытать растерянность без четкого разграничения заказов по исполнителям, клиентам и времени изготовления и доставки заказа.

Необходимость автоматизации бизнес-процесса имеет убедительное обоснование.

В основе внедрения ИС в организации всегда лежат некоторые целевые установки. Примером могут служить сокращение сроков обработки информации, повышение прозрачности бизнеса, экономия временных и материальных ресурсов и т.п. Информационные системы, эксплуатируемые на современных предприятиях, это внушительные программно-аппаратные комплексы, имеющие сложный состав, включающий в себя интерфейсные и функциональные элементы, модули, отчеты и другие важные компоненты.

В связи с этим к разработке автоматизированных информационных систем сложился определенный набор требований.

«Как правило, к разработке ИС определяют три уровня требований:

- уровень бизнес-требований, определяющий бизнес-результаты использования ИС,
- уровень пользовательских требований, призванный обеспечить удобство и простоту использования системы,
- уровень функциональных требований, регламентирующий поведение системы и ее функционирование» [15].

Сформулируем требования к информационной системе.

Перечислим функциональные требования к системе:

- информационная система должна поддерживать ведение списка заказов, списка постоянных заказчиков, списка продуктов, списка поставщиков;
- информационная система должна реализовывать функции добавления, удаления и редактирования записей в списках;
- система должна формировать Заказ и Лист назначений;
- определим нефункциональные требования к системе;
- графический интерфейс информационной системы должен быть удобным и интуитивно понятным;

- доступ ко всей информации и данным система должна предоставлять только авторизованным пользователям;
- любой пользователь системы (рядовой сотрудник компании) может получить доступ только к данным о назначении исполнителей на заказ;
- система должна быть простой в использовании и надежной.

Определив требования к системе, можно рассмотреть готовые решения, которые могли бы быть внедрены для автоматизации исследуемого бизнес-процесса.

1.3 Анализ существующих разработок на предмет соответствия сформулированным требованиям

1.3.1 Определение критериев анализа

Сегодня на рынке программного обеспечения предлагается большое количество программ для учета заказов. Среди профессионального программного обеспечения имеются как разработки от известных компаний-разработчиков, так и программы от менее знаменитых производителей.

Выполним анализ готовых решений, исходя из сформулированных ранее требований. В качестве критериев анализа выделим основные из требований, а также учтем стоимость лицензий на использование или пакетных предложений.

Критерии, по которым проведем сравнение готовых программных продуктов:

- удобство графического интерфейса;
- простота использования системы;
- гибкость настройки;
- организация базы данных заказов;
- автоматизация обработки заказов;

- наличие гарантийного и постгарантийного обслуживания;
- разграничение доступа к данным;
- стоимость лицензии.

1.3.2 Сравнительная характеристика существующих разработок

Выберем несколько подходящих для учета заказов ООО «Элит Эскорт» готовых программных продуктов и сравним их по выбранным критериям.

«1С:Розница» – универсальный вариант – разработка российской компании «1С», являющейся лидером производства программных продуктов для автоматизации бизнес-процессов на отечественном рынке.

Программный комплекс «1С: Розница» используют как небольшие торговые объекты, так представители крупных торговых сетей.

Функционал системы легко настраивается, позволяя учесть все нюансы бизнес-процессов конкретной организации и наладить максимально эффективную работу.

К функциям информационной системы «1С: Розница» относится:

- управление ценовой политикой и формирование ассортиментного ряда;
- составление заявок поставщикам на основе контроля закупок и анализа остатков;
- обработка нормативной и справочной документации;
- поддержка комплекса маркетинговых мероприятий;
- учет запаса товаров на складе;
- сбор статистических и аналитических данных.

Стоимость программы для 1 рабочего места (пакетный вариант) составляет 3600 руб.

Программа Delans Менеджер подходит для любого бизнеса, который занимается доставкой и перевозками.

Пример рабочего окна Delans Менеджер представлен на рисунке 8.

Дата созда.	Номер наклад.	Вх наклад.	Статус	Дата доставки	Отправитель	Контактное лицо	Телефон	Город	Получатель
24.04.2020	0000000000780	551146	Создан	25.04.2020	Санкт-Пе...	А. Михаил Серг...			
24.04.2020	0000000000781	551261	Создан	25.04.2020	Санкт-Пе...	Андросов Род...			
24.04.2020	0000000000782	551174	Выполняется	25.04.2020	Санкт-Пе...	Аркасов Серге...			
24.04.2020	0000000000783	551174	Выполняется	25.04.2020	Санкт-Пе...	Аркасов Серге...			
24.04.2020	0000000000784	551174	Выполняется	25.04.2020	Санкт-Пе...	Аркасов Серге...			
24.04.2020	0000000000785	551215	Выполняется	25.04.2020	Санкт-Пе...	Веденева Ан...			
24.04.2020	0000000000786	551223	Выполняется	25.04.2020	Санкт-Пе...	Гехеева Марат...			
24.04.2020	0000000000787	551270	Выполняется	25.04.2020	Санкт-Пе...	Герасимов Се...			
30.04.2020	0000000000820	551174	Выполняется	01.05.2020	Санкт-Пе...	Аркасов Серге...			
30.04.2020	0000000000822	551146	Выполняется	01.05.2020	Санкт-Пе...	А. Михаил Серг...			
30.04.2020	0000000000821	551215	Частично вып.	01.05.2020	Санкт-Пе...	Веденева Ан...			
30.04.2020	0000000000823	551223	Частично вып.	01.05.2020	Санкт-Пе...	Гехеева Марат...			
30.04.2020	0000000000824	551261	Создан	01.05.2020	Санкт-Пе...	Андросов Род...			
30.04.2020	0000000000825	551270	Создан	01.05.2020	Интернет магази...	Слабинский Светос...			

Рисунок 8 – Окно обработки заказов Delans Менеджер

Специалисты, для которых Delans Менеджер будет полезен: руководитель, менеджер по работе с клиентами, логист, диспетчер, финансовый сотрудник, сотрудник склада.

Функции для руководителя:

- сквозная аналитика – деятельность компании и логистика;
- взаимодействие и обмен данными между подразделениями;
- снижение уровня нагрузки на сотрудников;
- уменьшение расходов на ПО и оборудование.

Функции для диспетчера:

- онлайн контроль выполнения плана доставки;
- отслеживание местоположения курьера;
- быстрый обмен данными с курьерами;
- управление и контроль равномерности загрузки курьеров.

Стоимость лицензии на использование программного обеспечения составляет 2850 руб./ месяц для одного пользователя.

RetailCRM – омниканальная система управления заказами и клиентами для интернет-магазинов и ритейлеров в едином окне (рисунок 9).

IP-телефония: звонки автоматически распределяются между всеми менеджерами. При входящем звонке отображается всплывающая карточка звонка с краткой информацией о клиенте, заказе и его статусе. Вся история и

запись звонков доступна в карточке заказа и логе звонков. Назначение задач менеджерами. Присутствует подключение всевозможных коммуникационных каналов: e-mail, sms, социальные сети, мессенджеры.

Подписка от 1500 руб./год за одного пользователя. Имеется бесплатная версия для малого бизнеса на 1 пользователя (300 заказов в месяц).

Номер	Дата и время	Тип клиента	Покупатель	Тип	Менеджер	Статус заказа	Оплата	Сумма
70872C	02.12.2020 09:34	Клиент	Григорий	Физическое лицо	Ангелина	Выполнен	✓	4 516,29 Р
70871C	28.11.2020 15:50	Корпоративный Клиент	ООО ОблТРАНС (Анна)	Юридическое лицо	Виктор	Новый	⌚	6 038, Р
70870C	28.11.2020 10:29	Клиент	Алексей Попов	Физическое лицо	Владимир	Отменен	⌚	10 516,40 Р
70869C	27.11.2020 19:50	Клиент	Петров Игорь	Физическое лицо	Анна	Комплект...	✓	75 070,00 Р
70868C	27.11.2020 16:20	Корпоративный Клиент	ООО Фарфор (Людмила)	Юридическое лицо	Антон	Доставка	⌚	42 516,29 Р
70867C	27.11.2020 15:34	Корпоративный Клиент	ООО Компания РемВосток (Гуляева Ульяна)	Юридическое лицо	Ангелина	Выполнен	✓	9 720,52 Р
70866C	27.11.2020 12:45	Клиент	Алексей Петров	Физическое лицо	Виктор	Согласова...	⌚	1 516,00 Р

Рисунок 9 – Окно обработки заказов RetailCRM

Сведем данные о рассмотренных программных продуктах в общую таблицу (таблица 2). Критерии оценены экспертами по 5-балльной системе (5-максимальный балл, 0-минимальный).

Таблица 2 – Соотношение времени заказа и времени доставки

Критерий	1С: Розница	Delans Менеджер	RetailCRM
Удобство графического интерфейса	5	3	3
Простота использования системы	1	4	3
Гибкость настройки	4	3	3
Организация базы данных заказов	4	3	4
Автоматизация обработки заказов	4	5	3
Наличие гарантийного и постгарантийного обслуживания	5	5	5
Разграничение доступа к данным	5	5	5
Стоимость лицензии	3	2	4
Среднее	3,875	3,75	3,75

Все программные продукты получили приблизительно одинаковую среднюю оценку. Небольшой перевес у информационной системы «1С: Розница». Но надо заметить, что ни одна из программ не набрала даже 4 баллов и ни одна не реализует в полной мере требования, предъявляемые к системе. В связи с этим принято решение о разработке собственного программного продукта.

1.4 Постановка задачи на разработку проекта создания информационной системы

Целью создания автоматизированной информационной системы управления заказами компании по доставке продуктов ООО «Элит Эскорт» является систематизация заказов, упорядочение деятельности по их обслуживанию, рациональное назначение персонала на выполнение работ.

Информационная система должна отвечать требованиям, сформулированным ранее, основными из которых являются:

- ведение базы данных (заказы, клиенты, продукты, поставщики);
- организация поиска информации по ключам;
- автоматизация обработки заказов;
- удобный, интуитивно понятный графический интерфейс пользователя;
- простота использования системы;
- гибкость настройки;
- разграничение доступа к данным.

Критерии эффективности системы:

- бесперебойная эффективная работа системы;
- время восстановления системы в случае отказа не более 30 минут.
- хранение в базе данных от 10000 записей;
- время обработки запроса не превышает 3 секунд;

– обслуживание до 15 заказов в течение 60 минут.

Для разработки системы может быть использована среда программирования Delphi7 (язык программирования Delphi). База данных может быть разработана в СУБД MS Access. В качестве языка запросов необходимо использовать язык структурированных запросов SQL.

1.5 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»

Совершенствование бизнес-процесса доставки продуктов ООО «Элит Эскорт» должно заключаться в систематизации заявок посредством автоматизации их разработки.

Принимая звонок по телефону, диспетчер будет вводить все данные в форму системы, что значительно ускорит время приема заявки (ориентировочно с 10-12 минут до 4 минут).

После получения заказа система формирует на основании введенных данных заказа три документа:

- Заказ – информация о заказе (номер, заказчик, дата и время приема, состав, адрес доставки, ориентировочное время доставки);
- Лист назначений – информация о назначении на заказ исполнителей (закупщика, упаковщика и курьера);
- Акт приема-передачи (стандартный, простая форма).

Сотрудники, исполняющие заказ, руководствуются сформированными документами.

Изменения, внесенные в модель в сравнении с моделью AS-IS (КАК ЕСТЬ), заключаются в добавлении в процесс автоматизированной информационной системы и правил ее использования.

Контекстная диаграмма бизнес-процесса TO-BE (КАК ДОЛЖНО БЫТЬ) и ее декомпозиции представлены на рисунках 10 – 12.

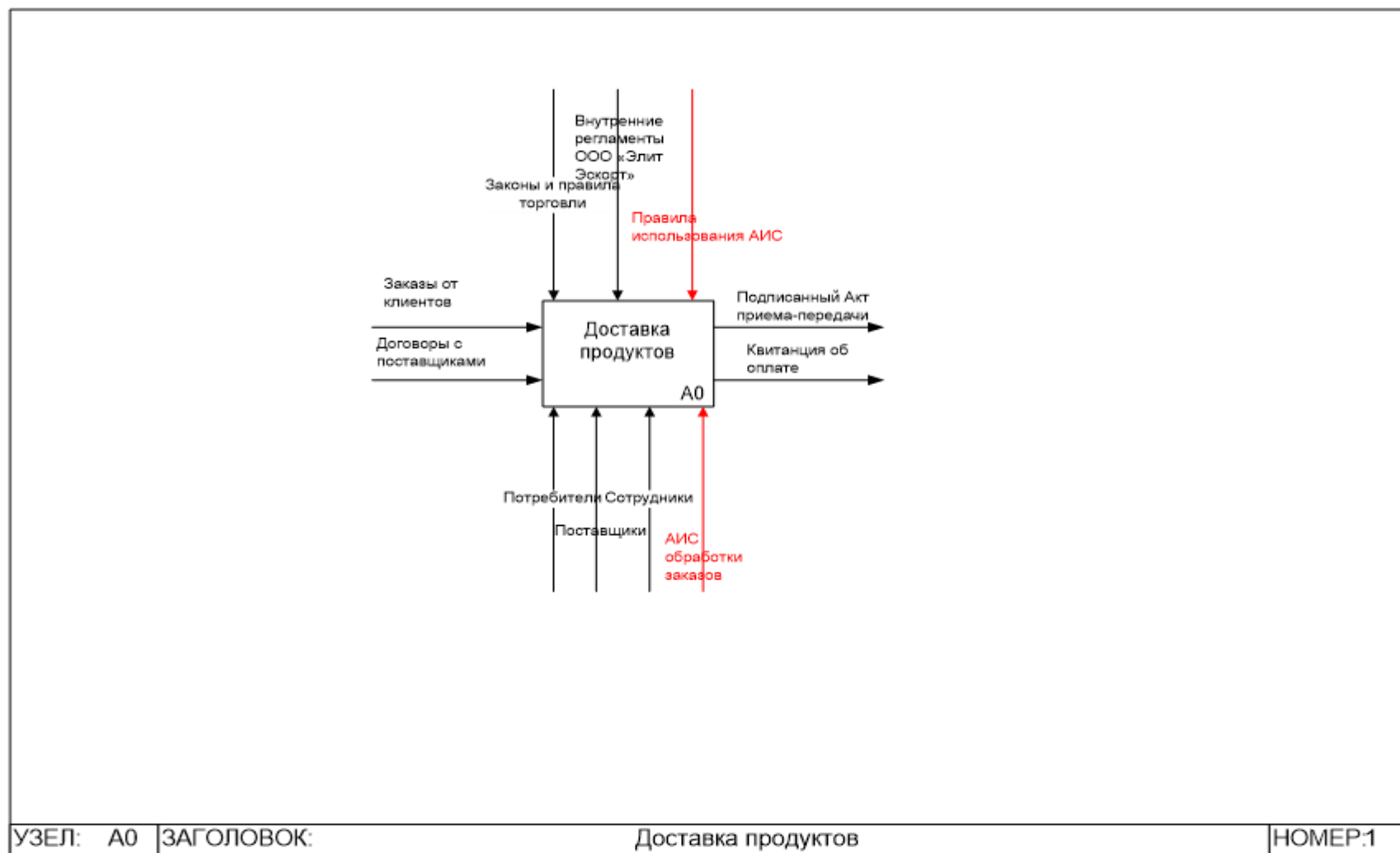


Рисунок 10 – Контекстная диаграмма ТО-ВЕ

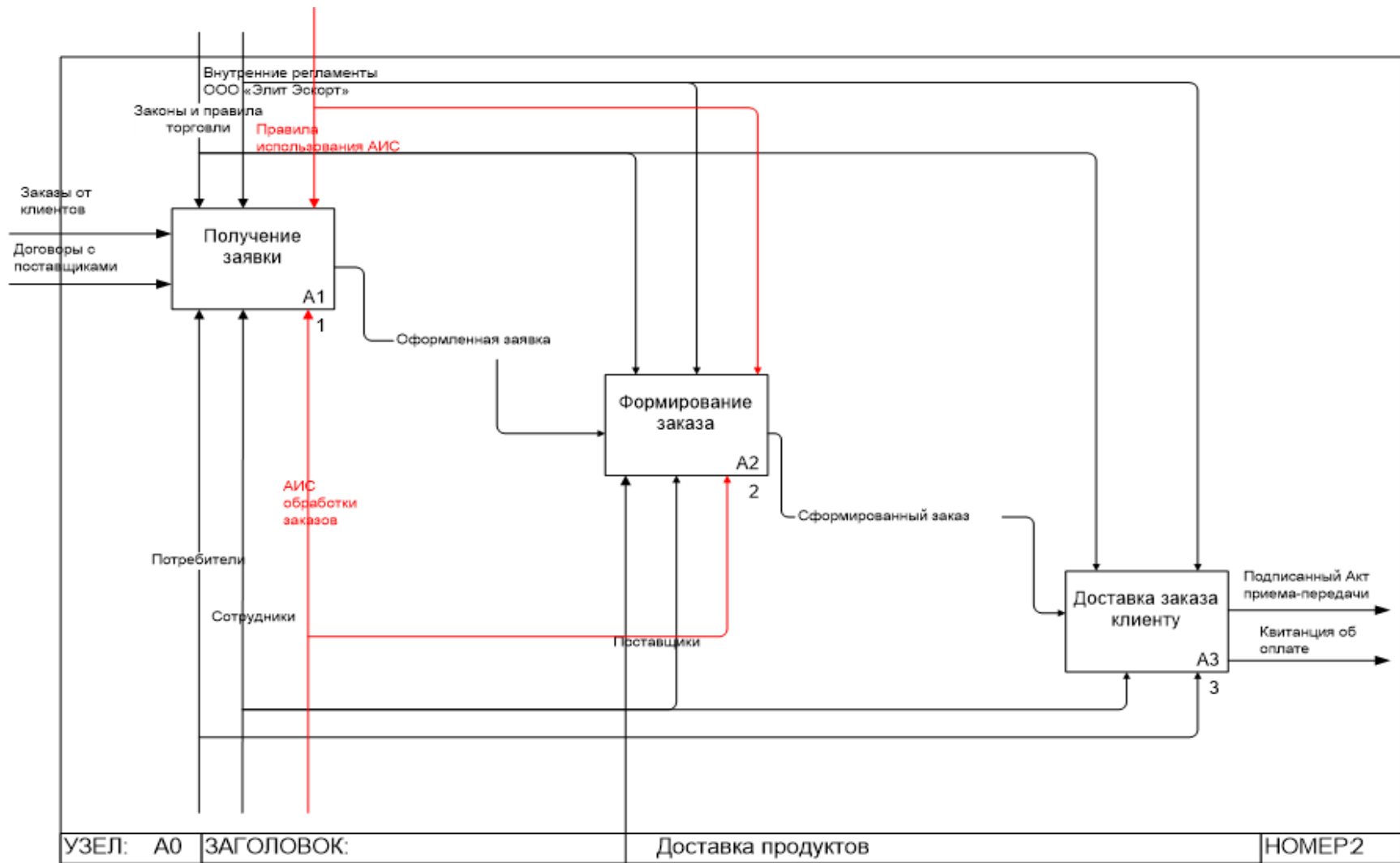


Рисунок 11 – Декомпозиция первого уровня бизнес-процесса ТО-ВЕ

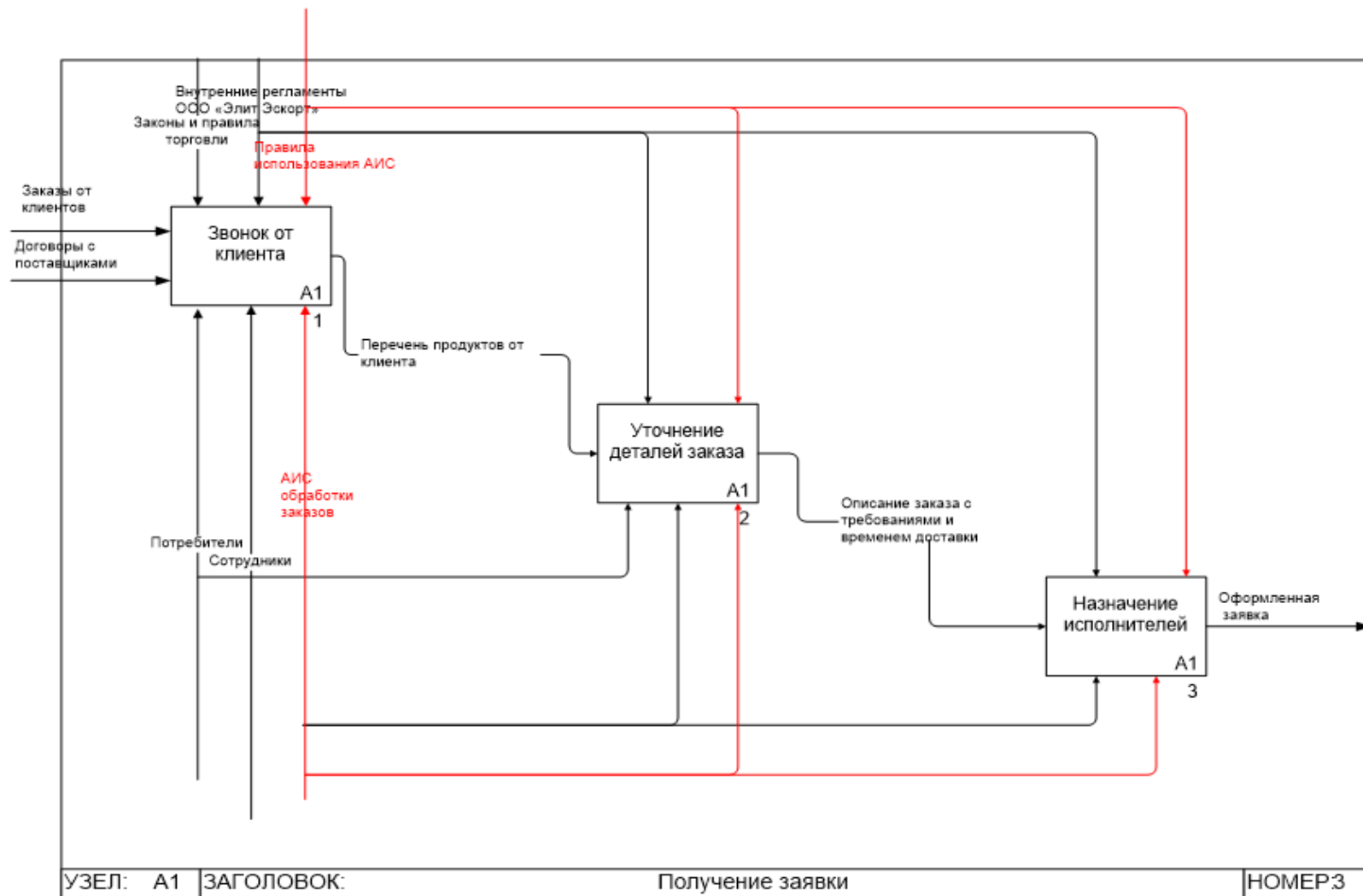


Рисунок 12 – Декомпозиция бизнес-процесса Получение заявки (ТО-ВЕ)

Выводы по главе 1

В первой главе выполнен анализ деятельности предприятия по доставке продуктов ООО «Элит Эскорт», представлены цели компании, описан основной бизнес-процесс. Описана и представлена схемой организационная структура предприятия.

Модель AS-IS основного бизнес-процесса представлена контекстной диаграммой и ее декомпозицией до второго уровня. Диаграммы представлены в нотации IDEF0. На основе модели КАК ЕСТЬ проанализированы уязвимые места бизнес-процесса в том представлении, которое имеется сегодня, и намечены пути его совершенствования посредством разработки информационной системы автоматизации обработки заказов. Построена модель КАК ДОЛЖНО БЫТЬ в нотации IDEF0.

Сформулированы требования к информационной системе и поставлена задача на ее разработку.

Глава 2 Логическое проектирование информационной системы

2.1 Выбор технологии логического моделирования

«Проектирование информационной системы представляет собой сложный процесс, который реализуют в несколько этапов. Количество этапов и их содержание определяются условиями и целями реализации проекта, а также профессиональными навыками команды разработки. Обычно выделяют следующие этапы проектирования ИС:

- анализ предметной области;
- проектирование системы;
- реализация системы;
- тестирование системы;
- ввод в действие, эксплуатация и сопровождение системы» [22].

Формирование модели данных осуществляют на этапе проектирования информационной системы. Исходными данными при этом являются результаты анализа предметной области. По результатам логического проектирования определяется концептуальная модель данных, формируется список входных данных, процессы, а также список выходных данных.

Логическое моделирование ИС можно реализовать с использованием всевозможных технологий как на основе объектно-ориентированного подхода, так и на основе функционального. В данной работе использован объектно-ориентированный подход. Для объектно-ориентированного подхода разработаны графические методы моделирования предметной области, обобщенные в языке унифицированного моделирования UML. Большинство существующих методов объектно-ориентированного проектирования ИС включают как язык моделирования, так и описание процесса моделирования. Для логического моделирования разрабатываемой

информационной системы выбран универсальный язык моделирования UML и одно из реализующих его CASE-средств – IBM Rational Rose [4].

2.2 Логическая модель АИС и ее описание

Логическая модель данных визуально отображает графическое представление структур данных, их атрибутов и связей. Логическая модель должна представить данные так, чтобы они могли быть естественно восприняты бизнес-пользователями. Проектирование логической модели должно быть независимо от требований платформы и языка реализации, а также способа дальнейшего использования данных.

Диаграмма вариантов использования представлена на рисунке 13.

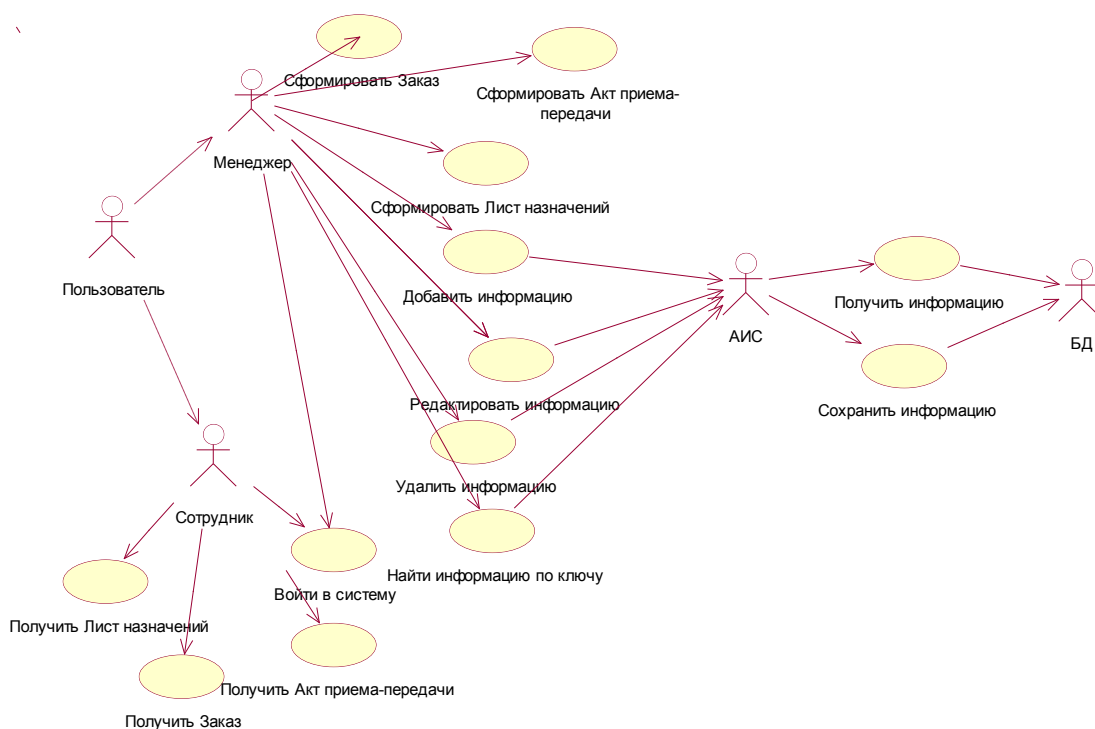


Рисунок 13 – Диаграмма вариантов использования

В системе предполагаются два типа пользователей: менеджер, сотрудник.

Под менеджерами понимаются директор ООО «Элит Эскорт» и диспетчеры. Эти пользователи системы имеют все права и могут выполнять следующие действия:

- добавлять в базу данных информацию о заказах, продуктах, поставщиках, заказчиках;
- редактировать в базе данных информацию о заказах, продуктах, поставщиках, заказчиках;
- удалять в базе данных информацию о заказах, продуктах, поставщиках, заказчиках;
- найти в базе данных информацию по ключу – номер заказа, ФИО заказчика, наименование продукта, наименование заказчика;
- формировать Заказ – документ с описанием времени получения заказа, количества и наименования продуктов, именем и адресом заказчика и временем вручения;
- формировать Лист назначений – документ с описанием состава заказа, временем доставки и назначением исполнителей.

Сотрудники – в данном контексте это обобщенное наименование курьеров, упаковщиков и закупщиков. Эта категория пользователей имеет доступ только к получению своих рабочих документов:

- получить Заказ;
- получить Лист назначений;
- получить Акт приема-передачи.

Диаграмма классов (class diagram) – диаграмма, предназначенная для представления модели статической структуры программной системы в терминологии классов объектно-ориентированного программирования. Диаграмма классов служит для представления статической структуры модели

системы в терминологии классов объектно-ориентированного программирования [3].

Диаграмма классов системы представлена на рисунке 14.

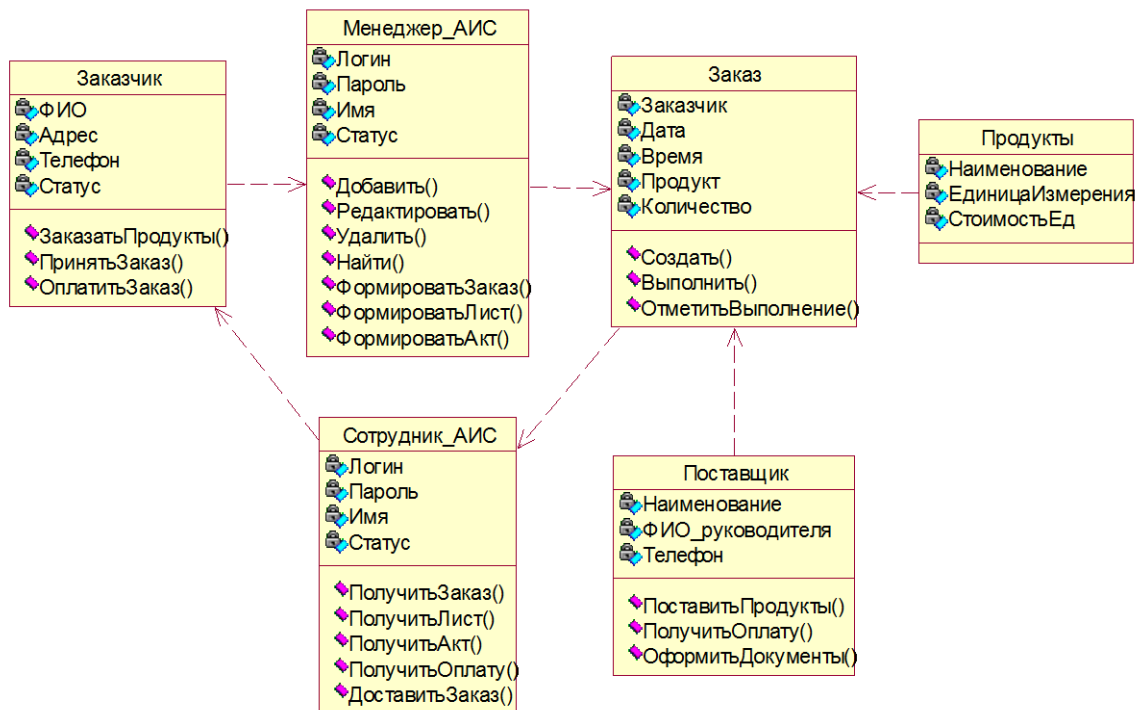


Рисунок 14 – Диаграмма классов системы

Информационно-логическая модель является основанием, на котором будет построена база данных системы.

2.3 Информационное обеспечение АИС

2.3.1 Используемые классификаторы и системы кодирования

Система классификации представляет собой совокупность правил и результат распределения заданного множества объектов на подмножества в соответствии с признаками сходства или различия.

«Классификация – логическая операция, состоящая в разделении всего изучаемого множества предметов по обнаруженным сходствам и различиям

на отдельные группы путем перечисления и указания признака или нескольких признаков, присущих члену группы.

Для решения конкретных задач в какой-либо области деятельности все множество объектов классифицируется по определенным признакам, и результаты такой работы фиксируются в классификаторах.

По степени обобщения классифицируемой информации различают четыре уровня деления:

- международный, применяемый в международных организациях, странах;
- общероссийский, используемый на уровне страны;
- отраслевой, действующий в пределах конкретной промышленности;
- локальный, используемый в пределах конкретного предприятия или группы предприятия» [22].

В разрабатываемой информационной системе используется локальный классификатор, использующийся в рамках ООО «Элит Эскорт», для определения статуса клиентов (постоянный/ не постоянный):

- 0 – не постоянный клиент;
- 1 – постоянный клиент.

Для определения группы поставки продуктов в информационной системе предусмотрен локальный классификатор, позволяющий наилучшим образом выбрать поставщика:

- 1 – универсальная группа (все продукты);
- 2 – овощи – фрукты;
- 3 – мясные и колбасные изделия;
- 4 – молочные продукты;
- 5 – хлебобулочные изделия;
- 6 – крупы;
- 7 – яйца;
- 8 – рыба и рыбные изделия;

9 – безалкогольные напитки;

10 – кондитерские изделия;

11 – бакалейные товары.

Справочник групп товаров может быть дополнен в процессе эксплуатации системы.

2.3.2 Характеристика нормативно-справочной и входной оперативной информации

«Под входной информацией имеют ввиду всю информацию, необходимую для решения задачи. Входная информация может быть размещена на различных носителях: первичных документах, машинных носителях, в памяти компьютера. Обобщая, можно считать, что входная информация – это данные, поступающие в информационную систему извне» [15].

При работе с проектируемой информационной системой можно выделить следующие объекты входной информации.

Справочник Поставщики содержит информацию о постоянных поставщиках ООО «Элит Эскорт»: наименование организации, фамилия, имя и отчество руководителя, телефон, группа поставки.

Макет экранной формы справочника Поставщики представлен на рисунке 15.

ПОСТАВЩИКИ

id Наименование

ФИО руководителя

Телефон Группа продуктов

поле вывода данных из БД

...

управляющие кнопки

Рисунок 15 – Макет экранной формы справочника Поставщики

Заполнение и дополнение справочника Поставщики происходит в процессе эксплуатации информационной системы. Справочник Заказчики предназначен для хранения информации о клиентах компании: фамилия, имя и отчество заказчика, адрес, телефон, статус (постоянный клиент – 1, не постоянный клиент – 0). Макет экранной формы справочника Заказчики представлен на рисунке 16.

The image shows a wireframe of a web form titled "ЗАКАЗЧИКИ". At the top, there are two input fields: "id" and "ФИО". Below them is a single wide input field for "Адрес", followed by another input field for "Телефон". A large rectangular box in the center is labeled "поле вывода данных из БД". At the bottom of the form, there are several small rectangular buttons, with an ellipsis "..." between two of them, labeled "управляющие кнопки".

Рисунок 16 – Макет экранной формы справочника Заказчики

Заполнение и дополнение справочника Заказчики происходит в процессе эксплуатации информационной системы.

Справочник Продукты предназначен для ведения списка продуктов, которые могут быть доставлены клиентам компании под заказ.

Справочник Продукты содержит следующую информацию: наименование, единицы измерения, стоимость единицы измерения, группа продуктов.

Справочник Группы продуктов предназначен для хранения информации о распределении продовольственных товаров по группам. Компания производит закупки разных групп товаров у разных поставщиков.

Справочник Группы продуктов вместе со справочниками Продукты и Поставщики позволяют сделать оптимальный выбор при комплектовании заказа.

Оперативная входная информация вводится оператором в процессе формирования заказа:

- ФИО заказчика (может быть введено из списка – новый заказчик или выбрано из имеющего списка);
- список продуктов и их количество;
- отметка о выполнении заказа.

Дата и время поступления заказа в систему также являются входной информацией.

На основании входной информации может быть полностью сформирован и/или помечен как выполненный заказ, найдена нужная информация в базе данных.

2.3.3 Характеристика выходной информации

В процессе работы информационной системы управления заказами ООО «Элит Эскорт» должен быть сформирован набор выходных документов.

Основными выходными документами являются:

- Заказ;
- Лист назначений;
- Акт приема-передачи.

Кроме того, в процессе работы с системой оператор может получить сформированный в форме экранного текста результат выполнения запроса (функция поиск по ключу) к базе данных.

В Заказе отображается следующая информация:

- номер заказа;
- дата и время формирования заказа;
- ФИО заказчика;
- адрес заказчика;

- состав заказа в формате
- продукт – количество;
- ориентировочное время доставки.

Сформированный Заказ выводится в виде сообщения в приложении, а также сохраняется в виде текстового файла. Макет документа Заказ представлен на рисунке 17.

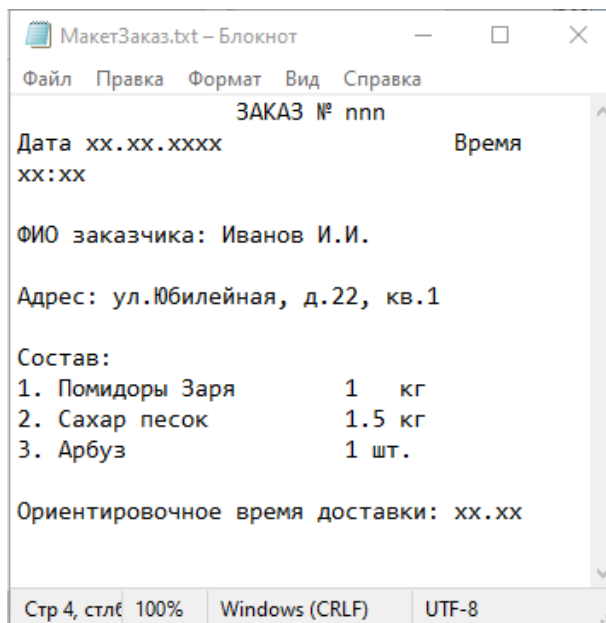


Рисунок 17 – Макет документа Заказ

Лист назначений формируется в информационной системе диспетчером сразу после создания заказа.

В Листе назначений отображается следующая информация:

- номер заказа;
- ФИО закупщика;
- ФИО упаковщика;
- ФИО курьера
- ФИО и телефон диспетчера.

Сформированный Лист назначения выводится в виде сообщения в приложении, а также сохраняется в виде текстового файла.

Макет документа Заказ представлен на рисунке 18.

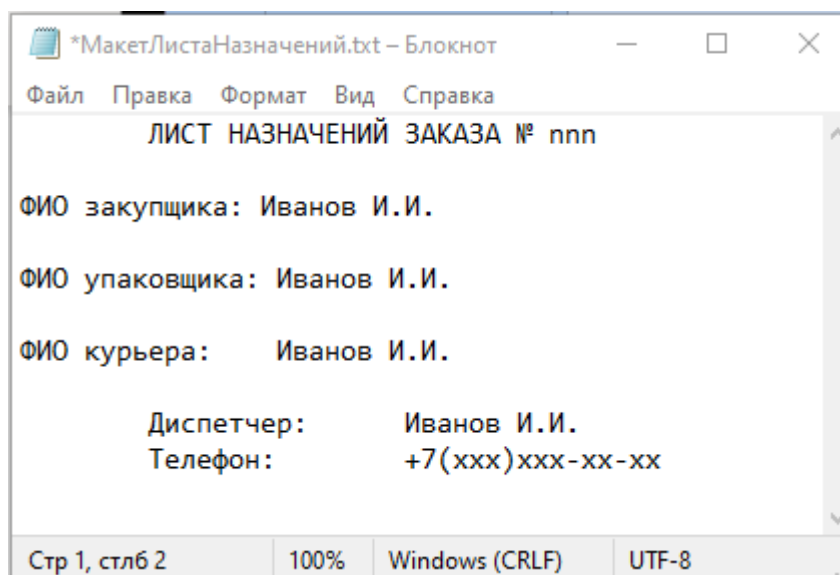


Рисунок 18 – Макет документа Лист назначения

Исполнители получают два документа – Заказ и соответствующий ему Лист назначений. Эти документы являются основанием для старта выполнения заказа исполнителями.

Акт приема-передачи представляет собой простую форму стандартного учетного документа, в котором указывают:

- номер заказа;
- дата и время формирования заказа;
- ФИО заказчика;
- адрес заказчика;
- состав заказа в формате:
- продукт – количество – цена;
- общая стоимость заказа с учетом доставки;
- время доставки.

Акт приема-передачи подписывает от ООО «Элит Эскорт» диспетчер, сформировавший заказ. Курьер после вручения заказа потребителю и проверки клиентом его полноты, целостности и качества, также получает в акте подпись клиента. Подписанный Акт приема-передачи символизирует завершение работы над заказом.

2.4 Проектирование базы данных информационной системы

2.4.1 Выбор технологии проектирования базы данных

Сущность структурного подхода к разработке информационных систем заключается в ее декомпозиции (разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи и т. д.

«Основные этапы, на которые разбивается процесс проектирования информационной системы, следующие:

- концептуальное проектирование – сбор, анализ и редактирование требований к данным (обследование предметной области, изучение ее информационной структуры, выявление всех фрагментов, каждый из которых характеризуется пользовательским представлением, информационными объектами и связями между ними, процессами над информационными объектами, моделирование и интеграция всех представлений);
- логическое проектирование – преобразование требований к данным в структуры данных. На выходе получают СУБД-ориентированную структуру базы данных и спецификации прикладных программ;
- физическое проектирование – определение особенностей хранения данных, методов доступа и т.д.

Логическая структура базы данных – это описание состава, типа и длины информационных единиц базы данных и связей между ними.

Сущности и связи модели данных представляются в виде реляционной таблицы (отношения). Отношение, соответствующее сущности, содержит атрибуты (столбцы), являющиеся атрибутами сущности и описывающие сущность (объект). Атрибут или множество атрибутов, которые однозначно определяют объект, называются первичным ключом.

Удобно представлять отношение как таблицу, где каждая строка есть кортеж, и каждый столбец соответствует одному компоненту. Столбцы при этом называются атрибутами и им присваивают имена. Список имен атрибутов называется схемой отношения. Совокупность схем отношений, используемых для представления информации, называются схемой базы данных, а текущие значения соответствующих отношений – базой данных.

Логический уровень представления модели – это абстрактный взгляд на данные, на нем данные представляются так, как выглядят в реальном мире. Логическая модель данных является универсальной и никак не связана с конкретной реализацией СУБД» [17].

Для проектирования базы данных выбраны табличное описание сущностей и их атрибутов и построение ER-модели. Построение ER-модели и логической модели базы данных АИС выполнено с использованием CASE-средства MS Visio.

2.4.2 Разработка концептуальной модели данных

Сущность – класс однотипных объектов, информация о которых имеет существенное значение для рассматриваемой предметной области.

На основании выполненного ранее анализа предметов в проектируемой базе данных можно выделить следующие основные сущности:

- Заказчик;
- Поставщик;
- Заказ.

Также для оптимальной организации базы данных понадобятся следующие дополнительные сущности:

- Продукт;
- Группа продуктов;
- Поставка.

Диаграмма «сущность- связь» (ER-модель) представлена на рисунке 19.

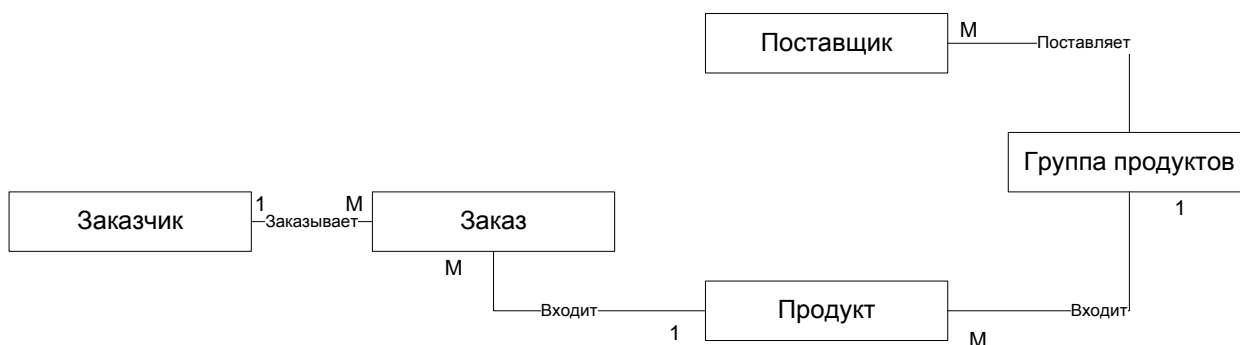


Рисунок 19 – Диаграмма «сущность- связь»

Все связи в проектируемой базе данных имеют связь «один-ко многим», что позволяет обеспечить целостность хранимых и обрабатываемых в БД данных. Для каждой сущности характерным является набор атрибутов. Набор атрибутов каждой сущности с выделением уникального атрибута, который может использоваться в качестве первичного ключа, и атрибутов для связи между таблицами (могут быть использованы как внешние ключи) представлен в таблицах 3 – 7.

Таблица 3 – Атрибуты сущности Заказчик

Наименование атрибута	Тип данных	Примечания
Код заказчика	целое число	первичный ключ
ФИО	текстовые данные	
Адрес	текстовые данные	
Статус	целое число	ограничения: 0 или 1
Телефон	текстовые данные	

Таблица 4 – Атрибуты сущности Поставщики

Наименование атрибута	Тип данных	Примечания
Код поставщика	целое число	первичный ключ
Наименование	текстовые данные	
ФИО руководителя	текстовые данные	
Телефон	текстовые данные	
Группа продуктов	целое число	внешний ключ, ограничения 0 - 10

Таблица 5 – Атрибуты сущности Продукты

Наименование атрибута	Тип данных	Примечания
Код продукта	целое число	первичный ключ
Наименование	текстовые данные	
Ед. измерения	текстовые данные	
Стоимость за ед.	действительное число	
Группа продуктов	целое число	внешний ключ, ограничения 0 - 10

Таблица 6 – Атрибуты сущности Группа продуктов

Наименование атрибута	Тип данных	Примечания
Код продукта	целое число	первичный ключ
Наименование	текстовые данные	

Таблица 7 – Атрибуты сущности Заказ

Наименование атрибута	Тип данных	Примечания
Код заказа	целое число	первичный ключ
Дата	дата/ время	
Время	дата/ время	
Продукт	целое число	внешний ключ
Количество	целое число	
Отметка о выполнении	логическое значение	

На основании разработанной концептуальной модели может быть сформирована логическая модель базы данных.

2.4.3 Обоснование вида логической модели

Логическая модель базы данных АИС представлена на рисунке 20.



Рисунок 20 – Логическая модель базы данных АИС

Представленная логическая модель базы данных позволяет удовлетворить всем требованиям заказчика в части структуры данных и манипулирования данными.

2.5 Требования к аппаратно-программному обеспечению

Разрабатываемая информационная система должна быть многопользовательской и позволять одновременное обращение к ней нескольких пользователей. Существует несколько способов совместного использования базы данных Access в зависимости от потребностей и доступности ресурсов. В ООО «Элит Эскорт» используется одноранговая локальная сеть, которая позволяет организовать многопользовательский доступ к таблицам базы данных Access. Под одноранговой понимается сеть, каждый компьютер которой может предоставлять остальным подключенным к сети компьютерам доступ ко всем или некоторым своим ресурсам, то есть являться сервером и клиентом одновременно. Пример объединения компьютеров в одноранговую сеть представлен на рисунке 21.

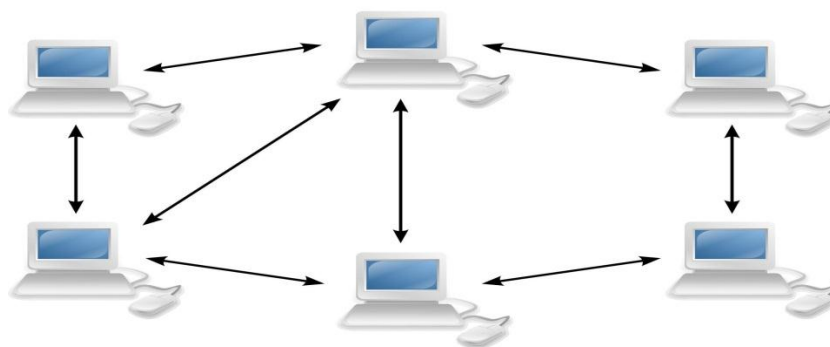


Рисунок 21 – Одноранговая компьютерная сеть

Одноранговая сеть может управляться встроенными компонентами настольных операционных систем версий Windows XP и выше. В ООО «Элит Эскорт» сеть работает под управлением ОС Windows 10. Один из компьютеров в сети выделен под файловый сервер. Такой подход ускоряет процесс обработки данных по сравнению с технологией, когда каждый узел в сети является клиентом и сервером одновременно. Требования, предъявляемые к серверу:

- частота процессора – не менее 3 ГГц;
- оперативная память – не менее 16 Гб;
- свободное место на жестком диске - не менее 1 Тб;
- подключение к сети Интернет;
- операционная система – Windows версии не ниже 7;
- наличие лазерного принтера с черно-белой печатью.

Требования, предъявляемые к рабочей станции:

- частота процессора – не менее 2.2 ГГц;
- оперативная память – не менее 4 Гб;
- свободное место на жестком диске - не менее 40 Гб;
- операционная система – Windows версии не ниже 7.

Требования к программному обеспечению:

- наличие установленной версии MS Access;
- наличие программы для вывода и печати текстовых файлов (возможно использование простейшего текстового редактора Блокнот).

Выполненный анализ предметной области и проектирование логической модели информационной системы являются основой для построения физической модели данных.

Выводы по главе 2

Во второй главе выполнен ряд работ, предшествующих физическому проектированию информационной системы:

- выполнен выбор технологии логического проектирования системы;
- разработаны диаграммы вариантов использования и классов;
- реализован проект логической модели АИС;
- представлены требования к аппаратно-программному обеспечению АИС.

Результатом работы является логическая модель информационной системы управления заказами компании «Элит Эскорт».

Глава 3 Физическое проектирование информационной системы

3.1 Выбор архитектуры информационной системы

Архитектура информационной системы описывает общую организацию системы, включая составляющие ее компоненты, существующие между ними отношения и связь системы с окружением.

Различают различные архитектуры информационных систем, выбор которых диктуется поставленными перед разработчиками АИС задачами и назначением системы.

В ВКР для реализации автоматизированной информационной системы выбрана архитектура «файл-сервер». Такая архитектура предполагает, что база данных размещается на сетевом сервере, а приложение клиента — на персональном компьютере пользователя (рисунок 22). Архитектура «файл-сервер» предполагает, что базу данных используют не более 10-15 пользователей одновременно. В процессе направления запросов база данных копируется на каждый персональный компьютер, а затем периодически обновляется.



Рисунок 22 – Архитектура «файл-сервер»

Выбор архитектуры обосновывается ее экономичностью, нет необходимости приобретать дорогостоящий мощный компьютер под сервер базы данных. Компания «Элит Эскорт» является небольшим предприятием, и база данных системы не потребует много места для хранения. Хотя ИС и планируется как многопользовательская система, одновременное количество запросов составляет не более 6 (количество компьютеров в одноранговой локальной сети организации).

3.2 Выбор технологии разработки программного обеспечения информационной системы

Задача разработки информационной системы предполагает этап выбора технологии разработки программного обеспечения. В настоящее время разработчикам предлагается более двух тысяч языков программирования высокого уровня. Активно используются на практике порядка 50. Выбор языка программирования обычно делают исходя из поставленных задач, необходимого функционала системы и опыта разработчика.

Для разработки информационной системы управления заказами компании по доставке продуктов ООО «Элит Эскорт» выбран язык программирования Delphi7, входящий в двадцатку наиболее популярных языков программирования по данным рейтинга ТЮВЕ (рисунок 23).

Oct 2021	Oct 2020	Change	Programming Language	Ratings	Change
1	3	▲	 Python	11.27%	-0.00%
2	1	▼	 C	11.16%	-5.79%
3	2	▼	 Java	10.46%	-2.11%
4	4		 C++	7.50%	+0.57%
...					
19	11	▼	 Perl	0.93%	-0.49%
20	22	▲	 Delphi/Object Pascal	0.93%	+0.22%

Рисунок 23 – Рейтинг ТЮВЕ за октябрь 2021 года (фрагмент)

Язык программирования Delphi достаточно прост в изучении. Он имеет много встроенных средств для организации привлекательного графического интерфейса и работы с базами данных, поддерживает структурированный язык запросов SQL.

Для реализации информационной системы выбран объектно-ориентированный подход. Язык программирования Delphi полностью поддерживает объектно-ориентированную концепцию программирования.

Разработка программ на языке Delphi может выполняться в нескольких средах разработки. В работе выбрана надежная устойчивая версия среды программирования – Delphi 7.

3.3 Выбор СУБД

Система управления базами данных необходима в разрабатываемой системе для администрирования базы данных ИС. В настоящее время разработчик может использовать большое количество различных СУБД, многие из которых являются реляционными, как и разрабатываемая в рамках проекта база данных.

Выбор СУБД осуществлялся в процессе сравнения среди популярных и доступных реляционных СУБД:

- MS Access,
- Oracle,
- MS SQL.

Критерии сравнения:

- удобство использования,
- интеграция с Delphi,
- стоимость,
- поддержка от разработчика,
- масштабируемость,
- простота освоения.

Результаты сравнения представлены в таблице 8.

Таблица 8 – Сравнение СУБД

Критерий	MS Access	Oracle	MS SQL
Удобство использования	+	+	-
Интеграция с Delphi	+	+	+
Стоимость	+	-	+
Поддержка от разработчика	+	+	+
Масштабируемость	+	+	+
Простота освоения	+	-	-
Всего:	6	5	4

Помимо того, что, с точки зрения разработчика, СУБД MS Access является оптимальным выбором для использования ее в разрабатываемой ИС, она также поддерживает архитектуру «файл-сервер», выбранную для реализации. Исходя из этого для разработки ИС принята СУБД MS Access.

3.4 Разработка физической модели данных

На основе представленной логической модели данных разрабатывается физическая модель данных, учитывающая особенности конкретной СУБД, в которой будет реализована база данных.

База данных информационной системы должна состоять из пяти таблиц, описывающих соответствующие сущности, описанные в логической модели: Заказчики, Заказы, Поставщики, Продукты, Группы продуктов. Описание структуры таблиц базы данных представлено в таблицах 9 – 13.

Таблица 9 – Описание структуры таблицы Заказчики

Поле	Тип данных	Описание
Код	счетчик	уникальное значение, первичный ключ
ФИО	текстовый	фамилия и инициалы заказчика
Адрес	текстовый	адрес заказчика
Статус	логический	постоянный/ первая покупка
Телефон	текстовый	номер мобильного телефона

Таблица 10 – Описание структуры таблицы Поставщики

Поле	Тип данных	Описание
Код	счетчик	уникальное значение, первичный ключ
Наименование	текстовый	наименование компании-поставщика
ФИО руководителя	текстовый	фамилия и инициалы руководителя компании-поставщика
Телефон	текстовый	номер мобильного телефона
Группа продуктов	числовой	внешний ключ

Таблица 11 – Описание структуры таблицы Продукты

Поле	Тип данных	Описание
Код	счетчик	уникальное значение, первичный ключ
Наименование	текстовый	название продукта
Ед_измерения	текстовый	единица измерения
Стоимость	денежный	стоимость единицы товара
Поставщик	числовой	внешний ключ

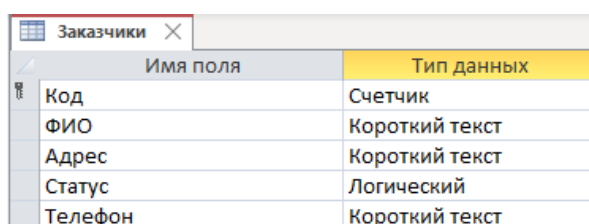
Таблица 12 – Описание структуры таблицы Заказы

Поле	Тип данных	Описание
Код	счетчик	уникальное значение, первичный ключ
Номер заказа	числовой	номер заказа в учетной системке компании
ФИО заказчика	числовой	внешний ключ
Дата	Дата и время	дата приема заказа
Время	Дата и время	время приема заказа
Продукт	числовой	внешний ключ
Количество	числовой	количество единиц заказанного товара

Таблица 13 – Описание структуры таблицы Группы продуктов

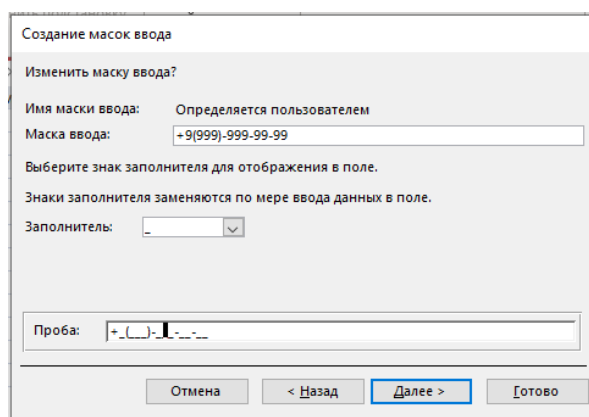
Поле	Тип данных	Описание
Код	счетчик	уникальное значение, первичный ключ
Наименование	текстовый	название группы продуктов

Все таблицы созданы в СУБД MS Access и представлены на рисунках 24 – 28.



Имя поля	Тип данных
Код	Счетчик
ФИО	Короткий текст
Адрес	Короткий текст
Статус	Логический
Телефон	Короткий текст

Рисунок 24 – Структура таблицы Заказчики



Создание масок ввода

Изменить маску ввода?

Имя маски ввода:

Маска ввода:

Выберите знак заполнителя для отображения в поле.

Знаки заполнителя заменяются по мере ввода данных в поле.

Заполнитель:

Проба:

Рисунок 25 – Маска для заполнения номера мобильного телефона

ГруппыПродуктов	
Имя поля	Тип данных
Код	Счетчик
Наименование	Короткий текст

Рисунок 26 – Структура таблицы Группы продуктов

Поставщики	
Имя поля	Тип данных
Код	Счетчик
Наименование	Короткий текст
ФИО	Короткий текст
Телефон	Короткий текст
ГруппаПродуктов	Числовой

Рисунок 27 – Структура таблицы Поставщики

Для заполнения поля Телефон использована такая же маска, как и в таблице Заказчики.

Для поля ГруппыПродуктов организована подстановка (рисунок 28, 29, 30).

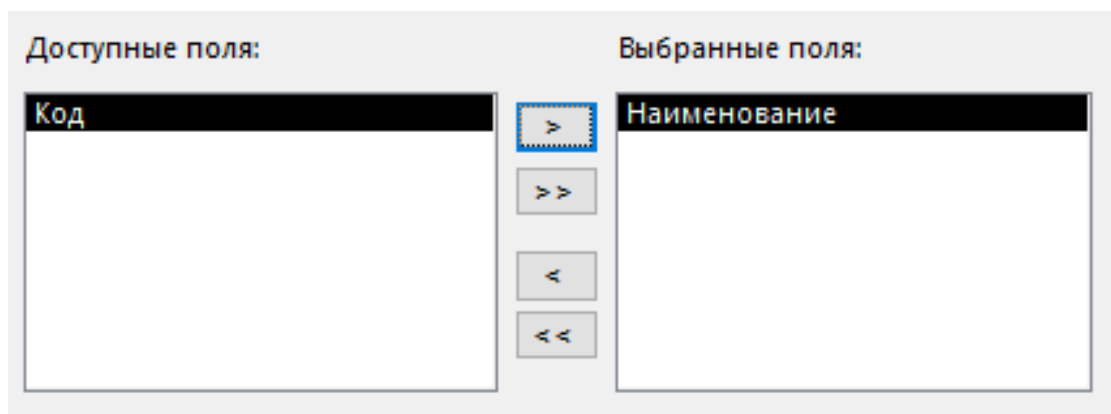


Рисунок 28 – Подстановка значений из таблицы ГруппыПродуктов в таблицу Поставщики

Продукты	
Имя поля	Тип данных
Код	Счетчик
Наименование	Короткий текст
Ед_измерения	Короткий текст
Стоимость	Короткий текст
Поставщик	Числовой

Рисунок 29 – Структура таблицы Продукты

Значения поля Поставщик – подстановка из таблицы Поставщики.

Имя поля	Тип данных
Код	Счетчик
Номер заказа	Числовой
ФИОЗаказчика	Числовой
Дата	Дата и время
Время	Дата и время
Продукт	Числовой
Количество	Числовой

Рисунок 30 – Структура таблицы Заказы

Схема данных базы данных представлена на рисунке 31.

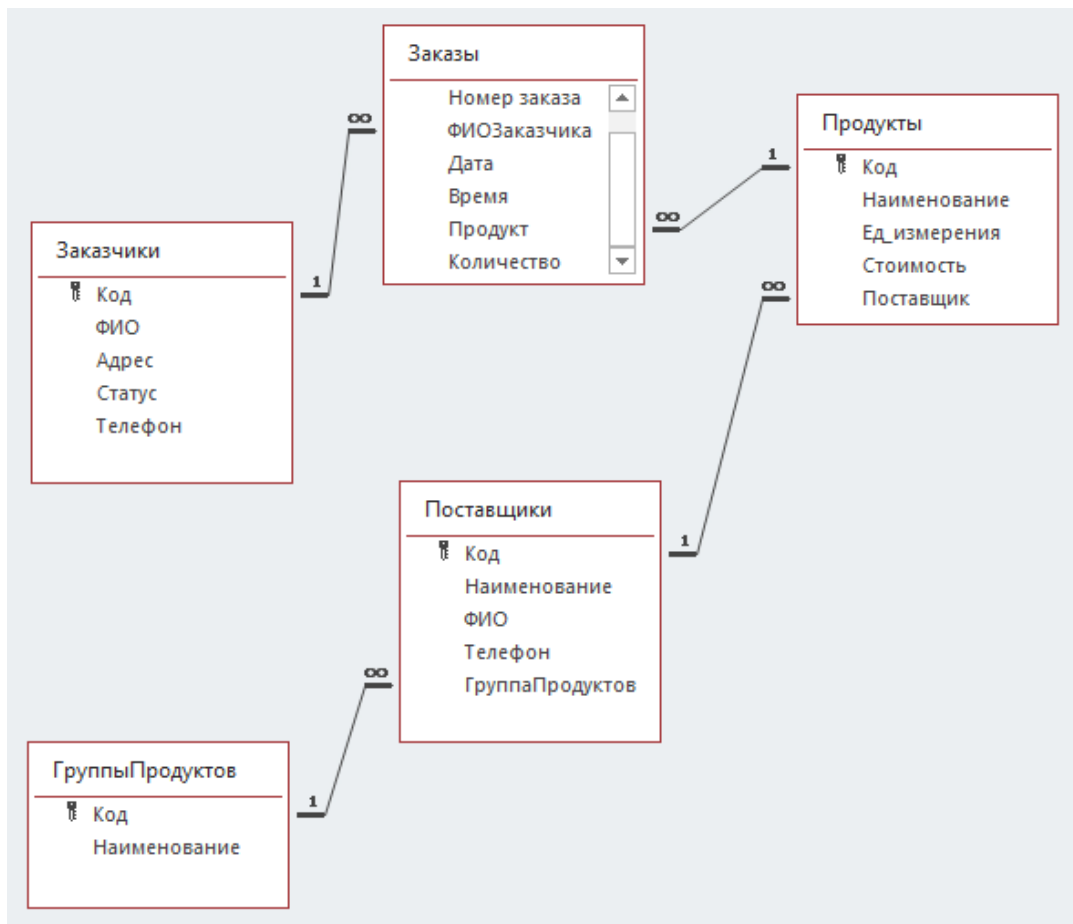


Рисунок 31 – Схема данных базы данных Orders

Все связи в базе данных имеют тип один-ко-многим. База данных готова к использованию в приложении.

3.5 Разработка программного обеспечения

3.5.1 Схема взаимодействия модулей информационной системы

Современное программное обеспечение решает множество сложных задач. Автоматизированные информационные системы, как правило, реализуют много различных функций, разработка которых обычно выделяется в отдельные наборы процедур. Это облегчает отладку и тестирование программы. Код выглядит логичнее, эстетичнее и профессиональнее.

Разработанная информационная система управления заказами фирмы по доставке продуктов «Элит Эскорт» является информационной системой, не требовательной к вычислительным мощностям и предназначена для обслуживания одновременно 6–10 человек, тем не менее она также имеет достаточно разветвленную структуру и состоит из различных модулей. Схема взаимодействия модулей представлена на рисунке 32.

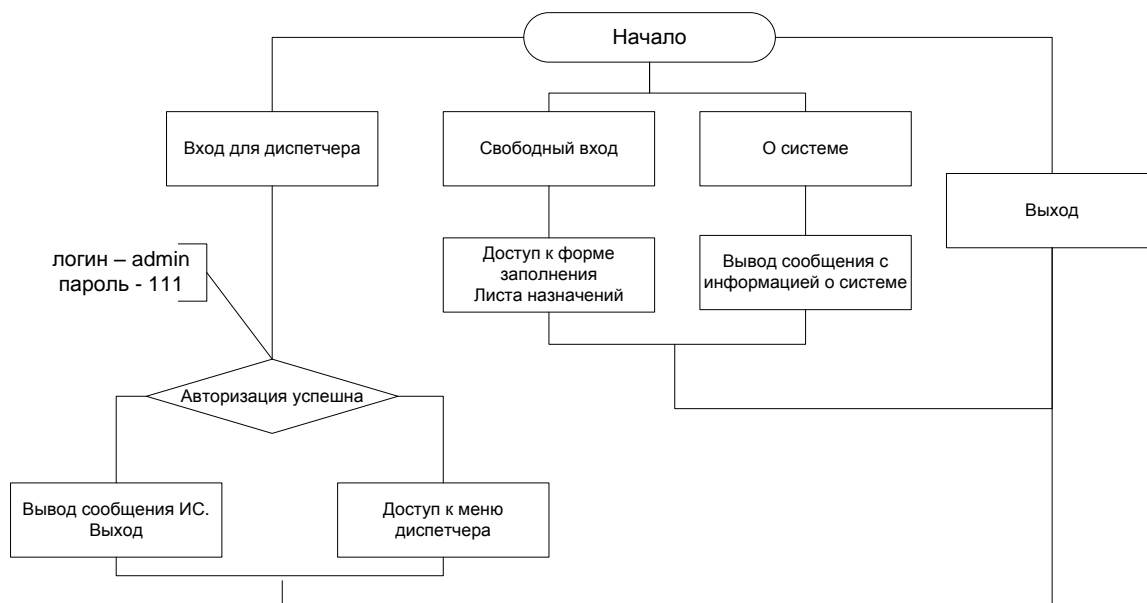


Рисунок 32 – Схема взаимодействия модулей приложения

Система компактная и, согласно требованиям заказчика, не слишком перегруженная сложным функционалом. С реализацией профессиональных задач в ИС должны справиться люди, не имеющие высшего образования и не являющиеся опытными пользователями, поэтому интерфейс системы минималистичен и рассчитан в первую очередь на простоту использования.

3.5.2 Описание модулей приложения

Приложение разработано с использованием объектно-ориентированного языка программирования Delphi7 и файл-серверной системы управления базами данных. Такой выбор был обусловлен тем, что компания «Элит Эскорт» является небольшой компанией и не готова нести

большие расходы на автоматизацию своего бизнеса. Но выбор также оказался и оптимальным, так как в организации имеется одноранговая компьютерная сеть. Установив сетевую версию СУБД Access, компания получила в свое распоряжение многопользовательскую информационную систему. Сегодня реализованы основные модули, но система будет развиваться, так как у нее имеется большой потенциал. Направлениями развития может стать как разработка новых модулей и функций, так и совершенствование имеющихся.

В разработанной информационной системе могут быть выделены следующие модули.

Входной модуль предназначен для информирования пользователей о назначении системы и разграничения прав доступа к системе. Хотя информационная система не содержит различных видов тайн, все же она обрабатывает один из видов конфиденциальной информации – персональные данные, поэтому доступ ко всем функциям системы разрешен только диспетчеру. Для входа в систему диспетчер должен указать логин и пароль. Идентификационные данные на время испытаний системы:

- логин: admin,
- пароль:111.

Входная форма системы представлена на рисунке 33.

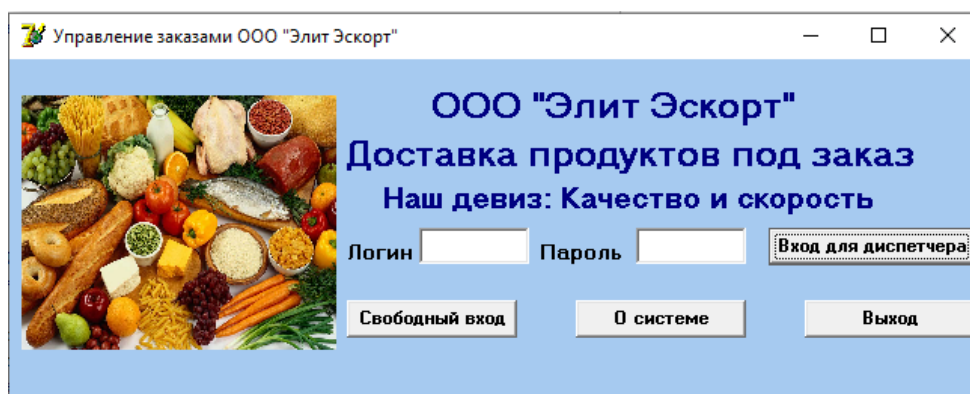


Рисунок 33 – Входная форма системы

Переход по кнопке «О системе» вызывает открытие справочно-информационной панели и не представляет интереса с точки зрения разработки ПО.

Кнопка «Выход» предназначена для безаварийного выхода из системы. Кнопка «Свободный вход» ведет на страницу листа назначений, к которому можно попасть и из аккаунта диспетчера, поэтому именно его систему проанализируем.

В основное меню системы менеджер и диспетчер попадают сразу после ввода своих регистрационных данных и входа в систему. Меню представлено на рисунке 34.



Рисунок 34 – Меню

Меню позволяет выбрать одно из четыре направлений – три по работе с данными из базы данных или назначение сотрудников на заказ и формирование листа назначения.

Инструменты для работы с базой данных размещены на одной форме, на разных вкладках многостраничного компонента (рисунок 35).

Управление заказами "Элит Эскорт". Администрирование данных

Заказчики | Поставщики | Продукты

Код	ФИО	Адрес	Статус	Телефон
3	Пантелеева Г.Н.	ул.Виноградная, д.11, кв.9	True	+7(923)-342-24-63
4	Романов Г.Д.	ул.Ленина, д.12, кв.11	False	+7(927)-643-52-22
5	Морозова А.А.	ул.Мира, д.33	True	+7(964)-864-82-24
6	Иванов И.И.	ул.Лозовая Аллея, д.9	False	+7(983)-342-41-11
10	Петровский А.Ю.	ул.Жукова, д.3, кв.17	True	+7(912)-342-22-11
11	Королев М.Т.	ул.Волжская, д.22	False	+7(912)-234-11-11
12	Молодцова Ч.К.	ул.Космонавтов, д.65, кв.11	False	+7(923)-342-33-10
13	Чткин М.А.	ул.Жукова, д.5, кв.15	True	+7(982)-231-64-53

Добавить Изменить Удалить Меню

Рисунок 35 – Страница Заказчики с тестовыми данными

Так как для разработки базы данных информационной системы использовалась СУБД MS Access, разработчиком было принято решение о разработке формы бланка заказов в среде этой СУБД. Access предлагает пользователю удобный и современный интерфейс и все инструменты для надежной и уверенной работы с базами данных.

3.6 Описание функциональности информационной системы

Информационная система предназначена для управления заказами на приобретение и доставку продуктов питания, поступающими в основном по телефону. Задача диспетчера – сформировать заказ и направить на его выполнение сотрудников, находящихся на рабочем месте.

Для введения информации в систему и ее редактирования возможно пользоваться как формами Access, так и приложением. Все изменения фиксируются «на лету» и сохраняются как в режиме непосредственной работы с СУБД «Access», так и фиксируются в приложении.

В системе реализованы следующие функции:

- добавление,

- удаление,
- редактирование записей БД,
- формирование бланка Заказа и Листа назначений.

Также в функции системы входит ведение справочников по основам направлениям обработки заказов – справочники поставщиков, заказчиков, продуктов и заказов.

Использование системы в перспективе позволит получить значительный эффект и выигрыш во времени.

Компоненты приложения для работы с базой данных представлены на рисунках 36 –37.

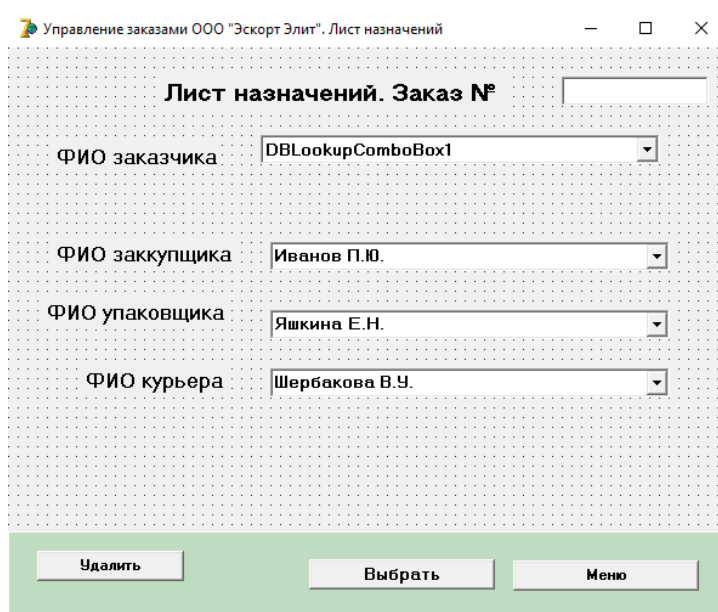


Рисунок 36 – Форма для формирования Листа назначений

В результате того, что все фамилии сотрудников внесены в раскрывающиеся списки, оформление листков назначения происходит очень быстро и оперативно.

Пример сформированного листка назначений представлен на рисунке 37.

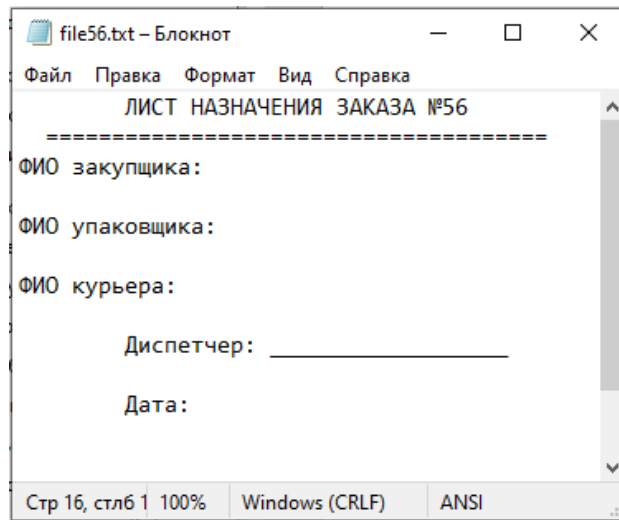


Рисунок 37 – Лист назначений

Также часть системы реализована как функции MS Access. На рисунке 38 представлена форма формирования заказов.

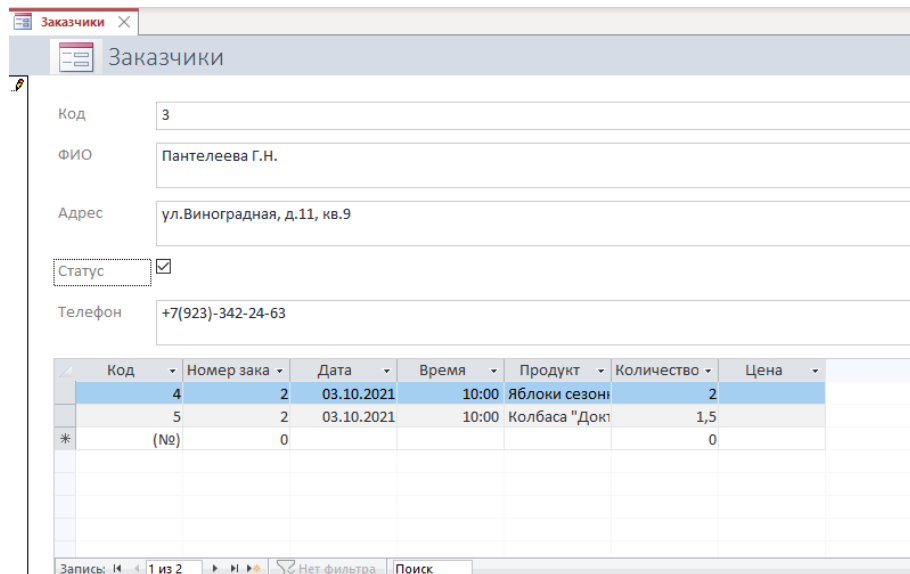


Рисунок 38 – Форма работы с заказами

Запрос на формирование бланка заказа представлен на рисунках 39-40. В процессе работы разработано два вида бланков заказов (рисунки 39–40).

Заказ № **2**

Фамилия: Пантелеева Г.Н. Дата: 03.10.2021 Ожидаемое время доставки: 15:30

Адрес: ул.Виноградная, д.11, кв.9

Наименование	Стоимость	Количество	ИТОГО
Колбаса "Докторская"	439	1,5	658,5
Яблоки сезонные	70	2	140
Всего:			798,5

Рисунок 39 – Яркий вариант бланка заказа

Заказ № **2** ООО "ЭЛИТ ЭСКОРТ"

Наименование Стоимость Кол-во ИТОГО

Фамилия: Пантелеева Г.Н. Дата: 03.10.2021

Адрес: ул.Виноградная, д.11, кв.9

Колбаса "Докторская"	439	1,5	658,5
Яблоки сезонные	70	2	140
Итого:			798,5

Рисунок 40 – Повседневный вариант бланка заказа

При форматировании бланков заполнение полей и вычисления осуществляются автоматически. Вручную заполняется только ориентировочное время доставки продуктового пакета.

3.7 Тестирование программного продукта

3.7.1 Выбор методов тестирования программного продукта

Объемы производимого и используемого в практических целях программного обеспечения заставляют всерьез задуматься о его качестве и

современных стандартах разработки ПО. На сегодняшний день разработано большое количество методов и типов тестирования программных продуктов.

Разработанная информационная система также прошла как отладку, так и тестирование – два важнейших этапа перед внедрением ИС в практику работы компании.

В тестировании приняли активное участие сотрудники предприятия-заказчика. Так как система небольшая, было принято решение тестировать ее в ручном режиме на основе разработанных требований.

3.7.2 Отладка и тестирование информационной системы

Отладка – это процесс обнаружения ошибок и их устранения. Отладка в процессе выполнения выпускной квалификационной работы выполнялась с помощью встроенных инструментов Delphi, в частности интерпретатора событий-ошибок Event Log. Пример сообщений, выдаваемых системой, представлен на рисунке 42.

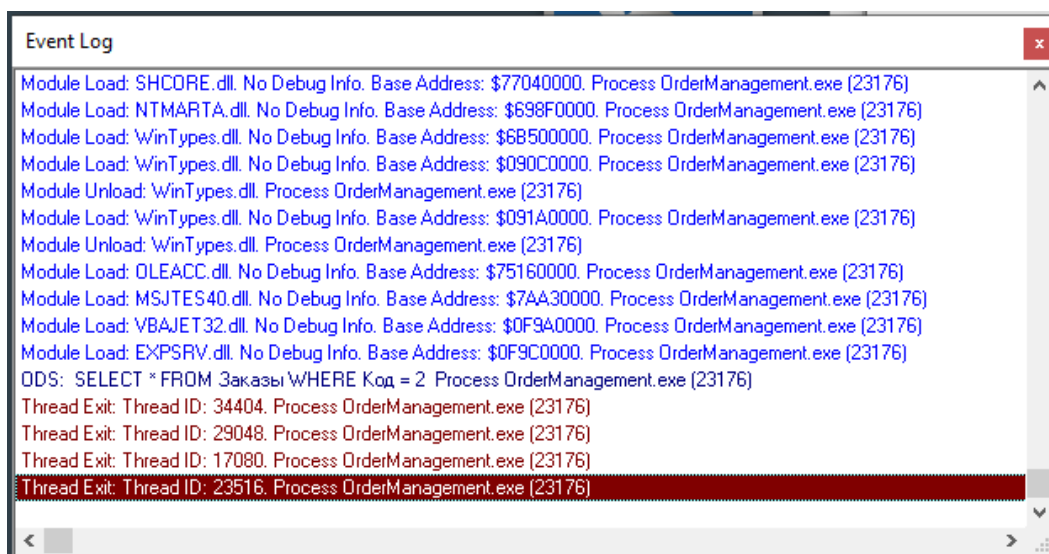


Рисунок 41 – Сообщения дебаггера

В процессе тестирования были проверены функции приложения с точки зрения полноты их реализации. По итогам тестирования были получены отдельные замечания и быстро устранены.

Программный код программного продукта полностью представлен в Приложении А.

По результатам тестирования сделаны выводы о том, что разработанная информационная система работает исправно и несет полезную информационную нагрузку.

Выводы по главе 3

В третьей главе рассмотрены вопросы архитектуры информационной системы, ее физического моделирования и реализации.

В качестве архитектуры информационной системы выбрана архитектура файл-сервер. Выбор обусловлен небольшими размерами компании и имеющейся IT-инфраструктурой.

Основным итогом выполнения третьей главы является практическая разработка выполненная на языке программирования Delphi и с применением СУБД Access.

Заключение

Процесс учета заказов на предприятиях любой сферы деятельности является одним из важнейших бизнес-процессов. От того, насколько оптимально и хорошо он реализуется, зависит качество выполнения работ, обслуживания клиентов, экономия времени и ресурсов компании и, в результате, ее прибыль. Как правило, процесс учета заказов хорошо подвергается автоматизации, поэтому разработка или внедрение автоматизированных информационных систем в целях оптимизации обслуживания заказов является грамотным и современным управленческим решением. В процессе выполнения выпускной квалифицированной работы была рассмотрена и решена задача автоматизации управления заказами компании по доставке продуктов на примере ООО «Эдит Эскорт».

Первым шагом в выполнении работы стало проведение исследования и изучение бизнес-процессов компании и определение места автоматизированной информационной системы в рамках существующей инфраструктуры и ресурсов организации. На основании полученного анализа предметной области было проведено логическое моделирование системы, выделение ее существенных частей и определение требований к системе. В процессе выбора системы разработки выбор был сделан в пользу языка программирования Delphi и системы управления базами данных MS Access, эти два программных продукта хорошо совместимы между собой. Кроме того, выбранное программное обеспечение оптимальное отвечает задаче разработки многопользовательской информационной системы для малого предприятия, оснащенного одноранговой локальной сетью.

Разработанная информационная система прошла тестирование, продемонстрировав корректную работу функций. В процессе тестирования также были намечены пути развития системы.

Список используемой литературы и используемых источников

1. Архангельский А.Я. Delphi. – М.: Бином, 2018. – 816 с.
2. Архангельский А.Я. Приемы программирования на Delphi на основе VCL. – М.Ж Бином, 2018. – 944 с.
3. Архангельский А.Я. Программирование в Delphi для Windows. – М.Ж Бином-пресс, 2017. – 1248 с.
4. Боровский А., Полищук Ю. Базы данных и их безопасность. Учебное пособие. – М.: Инфра-М, 2020. – 210 с.
5. Брантон С.Л., Куц Дж. Н. Анализ данных в науке и технике. – М.: ДМК Пресс, 2021. – 542 с.
6. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е изд. – М.: ДМК Пресс, 2006. – 496 с.
7. Звонарев, С. В. Основы математического моделирования: учебное пособие / С. В. Звонарев. — Екатеринбург: Изд- во Урал. ун- та, 2019. – 112 с.
8. Канцедал С. – Алгоритмизация и программирование. – М.: Форум, 2018. – 352 с.
9. Леоненков А. В. Самоучитель UML 2. – СПб.: БХВ-Петербург, 2007. – 576 с.
10. Логачева М. Информационные системы и программирование. Администратор баз данных. – М: Инфра-М, 2020. – 439 с.
11. Мартишин С.А., Симонов В.Л., Храпченко М.В. Базы данных. Практическое применение СУБД SQL. – М.: Форум, 2019. – 368 с.
12. Петров А. Распределенные данные. Алгоритмы работы современных систем хранения информации. – СПб.: ВHV-Питер, 2021. – 336 с.
13. Попов И., Максимов Н., Голицына О. Базы данных: учебное пособие. – М.: Форум, 2019. 400 с.

14. Санников Е. Курс практического программирования в Delphi. Объектно-ориентированное программирование. Практикум. – М.: Солон-пресс, 2017. – 188 с.
15. Семенова О., Логачев М. Информационные системы и программирование. – М: Инфра-М, 2020. – 551 с.
16. Симонов В., Мартишин С., Храпченко М. Базы данных. – М.: Инфра-М, 2020. – 235 с.
17. Сьоре Э. Проектирование и реализация систем управления базами данных. – М.: ДМК-Пресс, 2021. – 466 с.
18. Тарасов С.В. СУБД для программиста. Базы данных изнутри. – М.: Солон-пресс, 2021. – 320 с.
19. Учебно-методическое пособие «Проектирование информационных систем» / Сост. Шамсутдинов Т.Ф. Казань: КГАСУ, 2018. – 110 с.
20. Шень А. Программирование: теоремы и задачи. – М.: МЦНМО, 2021. – 320 с.
21. Мочалин С. М., Шамис В. А. Управление заказами на предприятии с использованием автоматизированной CRM-системы // Научно-методический электронный журнал «Концепт». – 2016. – № 4 (апрель). – с.1 – 7.
22. Обеспечение интероперабельности при информационном моделировании объектов строительства. – М.: Федеральный центр нормирования, стандартизации и оценки соответствия в строительстве. – 158 с.

Приложение А

Код программы

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, jpeg;
type
  TForm1 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Image1: TImage;
    Button1: TButton;
    Label4: TLabel;
    Edit1: TEdit;
    Label6: TLabel;
    Edit2: TEdit;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    procedure Button4Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
implementation
{$R *.dfm}
//=====Выход=====
procedure TForm1.Button4Click(Sender: TObject);
begin
  Application.Terminate;
end;
//=====К системе=====
procedure TForm1.Button3Click(Sender: TObject);
var s:string;
begin
  s:= '-истема предназначена для управления заказами ккк
"Элит Эскорт"  '+#10+#13;
  s:=s+' диспетчер имеет доступ ко всем функциям. для
входа следует '+#10+#13;
  s:=s+' ввести логин и пароль и нажать кнопку выход для
диспетчера.'+#10+#13;
  s:=s+' кстальные сотрудникимогут распечатать бланк
заказа и лист назначений.'+#10+#13;
```

Продолжение Приложения А

```
        s:=s+' для входа следует нажать кнопку -ВОБОДНЫЙ
вход.'+#10+#13;
        s:=s+'                               'Хорошей работы!';
        MessageBox(0,PAnsiChar(s),'»нформация', MB_OK);
    end;
end.

unit Menu;

interface

uses

    Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms,
    Dialogs, StdCtrls;

type
    TForm2 = class(TForm)
        Label1: TLabel;
        Button2: TButton;
        Button1: TButton;
        Button3: TButton;
        Button4: TButton;
        Button6: TButton;
        procedure Button6Click(Sender: TObject);
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
        procedure Button3Click(Sender: TObject);
        procedure Button4Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form2: TForm2;

implementation
    Uses Main, Order, Data;
    {$R *.dfm}
    //=====Выход=====
    procedure TForm2.Button6Click(Sender: TObject);
    begin
        if MessageDlg('Выходим из программы?',
            mtConfirmation, [mbYes, mbNo], 0) = mrYes
        then
            begin
                Application.Terminate;
                Exit;
            end;
        end;
    end;
```

Продолжение Приложения А

```
        end;
    end;
    //=====Принять заказ=====
    procedure TForm2.Button1Click(Sender: TObject);
    begin
        Form3.Visible:=true;
        Form2.Visible:=false;
        Form3.Button5.Visible:= true;
    end;
    //=====Данные о поставщиках=====
    procedure TForm2.Button2Click(Sender: TObject);
    begin
        Form4.Visible:=true;
        Form2.Visible:=false;
        Form4.PageControl1.ActivePageIndex:=1;
    end;
    //=====Данные о продуктах=====
    procedure TForm2.Button3Click(Sender: TObject);
    begin
        Form4.Visible:=true;
        Form2.Visible:=false;
        Form4.PageControl1.ActivePageIndex:=2;
    end;
    //=====Данные о клиентах=====
    procedure TForm2.Button4Click(Sender: TObject);
    begin
        Form4.Visible:=true;
        Form2.Visible:=false;
        Form4.PageControl1.ActivePageIndex:=0;
    end;

end.

unit Order;

interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics,
    Controls, Forms,
    Dialogs, StdCtrls, DB, ADODB, Grids, DBGrids, ExtCtrls,
    DBCtrls;

type
    TForm3 = class(TForm)
        Panel1: TPanel;
        Button1: TButton;
        Button5: TButton;
        Label2: TLabel;
        Edit1: TEdit;
```

Продолжение Приложения А

```
Label4: TLabel;
DBLookupComboBox1: TDBLookupComboBox;
Label1: TLabel;
Label3: TLabel;
Label5: TLabel;
ComboBox1: TComboBox;
ComboBox2: TComboBox;
ComboBox3: TComboBox;
Button4: TButton;
Button6: TButton;
procedure Button1Click(Sender: TObject);
procedure FormActivate(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;

type
  OrderData = Record
    id: integer;
    date: string[10];
    time: string[10];
    clientFIO: string[50];
    basketTitles: array[0..10] of String;
    basketPrices: array[0..10] of Real;
    baskerQuantity: integer;
  end;

implementation
  Uses Menu, U1, DateUtils ;
  {$R *.dfm}
  //=====Меню=====
  procedure TForm3.Button1Click(Sender: TObject);
  begin
    Form2.Visible:=true;
    Form3.Visible:=false;
  end;
  //=====При загрузке формы=====
  procedure TForm3.FormActivate(Sender: TObject);
  //var
  //DateTime : TdateTime;
```


Продолжение Приложения А

```
begin
end;
//=====При показе формы=====
procedure TForm3.FormShow(Sender: TObject);

begin

//Показываем Заказы
DataModule1.ADOQZakaz.Close;
DataModule1.ADOQZakaz.SQL.Text:=      'SELECT      Заказы.Код,
Заказы.[Номер заказа], Заказы.ФИОЗаказчика, '
+'Заказчики.ФИО, Заказы.Дата, Mid(Заказы.Время, 1, 5) as
Время, Продукты.Наименование, Заказы.Количество '
+' FROM (Продукты INNER JOIN Заказы ON Продукты.Код =
Заказы.Продукт) '
+'      INNER JOIN      Заказчики      ON      Заказчики.Код      =
Заказы.ФИОЗаказчика;';

DataModule1.ADOQZakaz.Open;
//Задаем ширину колонок для DG2
{With DBGrid1 do
begin
Columns.Items[0].Width:=50;
Columns.Items[1].Width:=100;
Columns.Items[2].Width:=2;
Columns.Items[3].Width:=150;
Columns.Items[4].Width:=80;
Columns.Items[5].Width:=80;
Columns.Items[6].Width:=200;
end;
}

end;
//=====Меню=====
procedure TForm3.Button5Click(Sender: TObject);
begin
Form2.Visible:=true;
Form3.Visible:=false;
end;
//=====Выбрать=====
procedure TForm3.Button6Click(Sender: TObject);
var
f : TextFile;
s:string;
begin
//выбираем сотрудников для выполнения заказа

{проверяем, что комбобоксы заполнены}
if Edit1.Text='' then
begin
```

Продолжение Приложения А

```

        MessageBox(0, 'Укажите номер заказа', 'Информация',
MB_OK);
        Exit;
    end;
    s:=Edit1.Text;
    s:='file'+s+'.txt';
    AssignFile(f, s);
    Rewrite(f);
    {запоминаем все в текстовый файл}
    s:='';
    s:='          ЛИСТ НАЗНАЧЕНИЙ ЗАКАЗА № '+ Edit1.Text+
(#13#10);

s:=s+'=====
'+ (#13#10);
        s:=s+ '    Закупщик - ' + ComboBox1.Text+(#13#10);
        s:=s+ '    Сборщик - ' + ComboBox2.Text+(#13#10);
        s:=s+ '    Курьер - ' + ComboBox2.Text+ (#13#10);
        s:='          Диспетчер _____'+
(#13#10);
        s:='          Дата:
_____'+(#13#10);

        Writeln(f, s);
        CloseFile(f);
    MessageBox(0, PChar(s), 'Информация', MB_OK);
    { if CheckBox1.Checked then
begin
        DBLookupComboBox1.Visible:=true ;
        Edit2.Visible:=false;
    end
    else
begin
        Edit2.Visible:=true;
        DBLookupComboBox1.Visible:=false;
    end;
    Label4.Visible:=true;
end;

// функция отдаёт
procedure getOrderDate(OrderID: Integer);
// переменная для данных заказа
var order : OrderData;
begin

        //Закрываем Поставщиков
        DataModule1.ADOQZakaz.Close;

        DataModule1.ADOQZakaz.SQL.Text:=' SELECT * FROM Заказы
WHERE Код = ' + IntToStr(OrderID);

```

Продолжение Приложения А

```
OutputDebugString( PChar(DataModule1.ADOQZakaz.SQL.Text) );

DataModule1.ADOQZakaz.Active := True;
ShowMessage(DataModule1.ADOQZakaz.Fields[1].AsString);

{
  DataModule1.ADOQZakaz.Open;

  while not DataModule1.ADOQZakaz.Eof do
  begin
    OutputDebugString( PChar('Дата: ' +
DataModule1.ADOQZakaz.FieldByName('Дата').AsString) );
    // OutputDebugString();
    // OutputDebugString();
    end;

    //Показываем Поставщиков
    DataModule1.ADOQZakaz.Close;
  }
end;

procedure TForm3.Button2Click(Sender: TObject);
begin
  {getOrderDate(2); }
end;
end.

unit Data;

interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, Grids, DBGrids, ExtCtrls, StdCtrls, ComCtrls,
  Mask, DBCtrls, DB;

type
  TForm4 = class(TForm)
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    TabSheet3: TTabSheet;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    Button1: TButton;
    Panel1: TPanel;
    Panel2: TPanel;
```

Продолжение Приложения А

```
Button5: TButton;  
Button6: TButton;  
Button7: TButton;  
Button8: TButton;  
Panel3: TPanel;  
Button9: TButton;  
Button10: TButton;  
Button11: TButton;  
Button12: TButton;  
Panel5: TPanel;  
Button13: TButton;  
Button14: TButton;  
Button16: TButton;  
Button17: TButton;  
Button18: TButton;  
Button19: TButton;  
Button20: TButton;  
DBGrid1: TDBGrid;  
DBGrid3: TDBGrid;  
Label1: TLabel;  
Edit1: TEdit;  
Label2: TLabel;  
Edit2: TEdit;  
Label3: TLabel;  
Edit3: TEdit;  
Label4: TLabel;  
CheckBox1: TCheckBox;  
Label9: TLabel;  
Label10: TLabel;  
Label11: TLabel;  
CheckBox3: TCheckBox;  
Label12: TLabel;  
Edit9: TEdit;  
MaskEdit2: TMaskEdit;  
DBGrid2: TDBGrid;  
Panel4: TPanel;  
Label5: TLabel;  
Label7: TLabel;  
Label8: TLabel;  
Edit4: TEdit;  
Edit5: TEdit;  
Edit6: TEdit;  
Label6: TLabel;  
DBLookupComboBox1: TDBLookupComboBox;  
Button21: TButton;  
Button22: TButton;  
Button23: TButton;  
Button24: TButton;  
Label13: TLabel;  
procedure Button20Click(Sender: TObject);
```

Продолжение Приложения А

```
procedure FormCreate(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure Button16Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button17Click(Sender: TObject);
procedure Button21Click(Sender: TObject);
procedure Button18Click(Sender: TObject);
procedure Button19Click(Sender: TObject);
procedure Button13Click(Sender: TObject);
procedure Button24Click(Sender: TObject);
procedure Button14Click(Sender: TObject);
procedure Button23Click(Sender: TObject);
procedure DBGrid2CellClick(Column: TColumn);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form4: TForm4;
  // идентификатор записи (строчки), по которой кликнули в
гриде
  // устанавливается по событию клика на ячейку грида
  currentSelectedRecordId : integer;

implementation
  Uses U1, Menu;
  {$R *.dfm}
  //=====Меню_Заказчики=====
==
procedure TForm4.Button20Click(Sender: TObject);
begin
  Form2.Visible:=true;
  Form4.Visible:=false;
end;
//===== При создании формы=====
procedure TForm4.FormCreate(Sender: TObject);
begin
  //Задаем ширину колонок для DG1
  With DBGrid1 do
  begin
    Columns.Items[0].Width:=30;
    Columns.Items[1].Width:=150;
    Columns.Items[2].Width:=200;
    Columns.Items[3].Width:=50;
    Columns.Items[4].Width:=100;
  end;
end;
```

Продолжение Приложения А

```
end;
//=====При показе формы =====
procedure TForm4.FormShow(Sender: TObject);
begin
  //Показываем Поставщиков
  DataModule1.ADOQPost.Close;
  DataModule1.ADOQPost.SQL.Text:='SELECT      Поставщики.Код,
Поставщики.Наименование as Наименование, '
  + '      Поставщики.ФИО,      Поставщики.Телефон,
ГруппыПродуктов.Наименование as ГруппаПродуктов '
  + 'FROM      ГруппыПродуктов INNER JOIN Поставщики ON
ГруппыПродуктов.Код = Поставщики.ГруппаПродуктов';
  DataModule1.ADOQPost.Open;
  //Задаем ширину колонок для DG2
  With DBGrid2 do
  begin
    Columns.Items[0].Width:=30;
    Columns.Items[1].Width:=200;
    Columns.Items[2].Width:=150;
    Columns.Items[3].Width:=100;
    Columns.Items[4].Width:=150;
  end;
  //Показываем Продукты
  DataModule1.ADOQProd.Close;
  DataModule1.ADOQProd.SQL.Text:='SELECT * FROM Продукты';
  {Продукты.Код, Продукты.Наименование, '
  + '      Продукты.Ед_измерения,      Продукты.Стоимость,
Поставщики.Наименование '
  + 'FROM Поставщики INNER JOIN Продукты ON Поставщики.Код =
Продукты.Поставщик;';}

  DataModule1.ADOQProd.Open;
  //Задаем ширину колонок для DG3
  With DBGrid3 do
  begin
    Columns.Items[0].Width:=30;
    Columns.Items[1].Width:=200;
    Columns.Items[2].Width:=150;
    Columns.Items[3].Width:=100;
    Columns.Items[4].Width:=150;
  end;
  //Задаем ширину колонок для DG1
  With DBGrid1 do
  begin
    Columns.Items[0].Width:=30;
    Columns.Items[1].Width:=150;
    Columns.Items[2].Width:=200;
    Columns.Items[3].Width:=50;
    Columns.Items[4].Width:=100;
  end;
end;
```

Продолжение Приложения А

```
end;

//=====Заказчика =====
procedure TForm4.Button16Click(Sender: TObject);
begin
    // Form2.Visible:=true;
    //Form4.Visible:=false;
end;

//=====Меню_Продукты=====
procedure TForm4.Button1Click(Sender: TObject);
begin
    Form2.Visible:=true;
    Form4.Visible:=false;
end;
//=====Добавить Заказчика=====
procedure TForm4.Button17Click(Sender: TObject);
begin
    //Добавляем пустую строку
    DataModule1.ADOTClient.Insert;
end;
//=====
procedure TForm4.Button21Click(Sender: TObject);
begin

end;
//=====Изменить
Заказчика=====
procedure TForm4.Button18Click(Sender: TObject);
begin
    if DataModule1.ADOTClient.Modified then
        DataModule1.ADOTClient.Post;
end;
//=====Удалить
Заказчика=====
procedure TForm4.Button19Click(Sender: TObject);
begin
    DataModule1.ADOTClient.Delete;
end;
//=====Готовим поле к добавлению
Поставщика=====
procedure TForm4.Button13Click(Sender: TObject);
begin
    Edit4.Text:='';
    Edit5.Text:='';
    Edit6.Text:='';
    Button24.Visible:=true;
    Button23.Visible:=false;
    DBLookupComboBox1.KeyValue:=1;
    Panel4.Visible:=true;
```

Продолжение Приложения А

```
end;
//=====Добавить запись Поставщик=====
procedure TForm4.Button24Click(Sender: TObject);
begin
    //Закрываем Поставщиков
    DataModule1.ADOQPost.Close;
    //Добавляем Поставщика
    DataModule1.ADOQPost.SQL.Text:='INSERT INTO Поставщики '
    +' ([Наименование], [ФИО], [Телефон], [ГруппаПродуктов]) '
    +' VALUES ("'+Edit4.Text+'", "'
    +Edit6.Text+'", "'
    +Edit5.Text+'", '
    +IntToStr(DBLookupComboBox1.KeyValue)+' )';
    DataModule1.ADOQPost.ExecSQL;

    //Обновляем таблицу Поставщиков
    DataModule1.ADOQPost.Close;
    DataModule1.ADOQPost.SQL.Text:='SELECT          Поставщики.Код,
Поставщики.Наименование, '
    +'          Поставщики.ФИО,          Поставщики.Телефон,
ГруппыПродуктов.Наименование '
    +' FROM      ГруппыПродуктов INNER JOIN Поставщики ON
ГруппыПродуктов.Код = Поставщики.ГруппаПродуктов;';
    DataModule1.ADOQPost.Open;
    With DBGrid2 do
    begin
        Columns.Items[0].Width:=30;
        Columns.Items[1].Width:=200;
        Columns.Items[2].Width:=150;
        Columns.Items[3].Width:=100;
        Columns.Items[4].Width:=150;
    end;
    Panel4.Visible:=false;
end;
//=====ГОТОВИМ          поле          к          ИЗМЕНЕНИЮ
Поставщика=====
procedure TForm4.Button14Click(Sender: TObject);
begin
    Edit4.Text:= DBGrid2.Fields[1].AsString; ;
    Edit6.Text:=DBGrid2.Fields[2].AsString;
    Edit5.Text:=DBGrid2.Fields[3].AsString;

    DataModule1.ADOTGroup.Locate ('Наименование', DBGrid2.Fields[4].AsString, [loCaseInsensitive, loPartialKey]);

    DBLookupComboBox1.KeyValue:=StrToInt (DataModule1.DataSGroup.DataSet.Fields[0].AsString);
    Button23.Visible:=true;
    Button24.Visible:=false;
```


Продолжение Приложения А

```

DBLookupComboBox1.KeyValue:=1;
Panel4.Visible:=true;
end;
//=====Изменяем запись
Поставщик=====
procedure TForm4.Button23Click(Sender: TObject);
begin
    //Закрываем Поставщиков
    DataModule1.ADOQPost.Close;
    //Изменяем Поставщика
    DataModule1.ADOQPost.SQL.Text:=' UPDATE Поставщики SET '
    +' Наименование = "'+ Edit4.Text+ '" '
    +' ,ФИО = "'+ Edit6.Text+'"'
    +' ,Телефон = "'+ Edit5.Text+'"'
    +' ,ГруппаПродуктов = ' + IntToStr(DBLookupComboBox1.KeyValue)
    +' WHERE Код = ' + IntToStr(currentSelectedRecordId);

    // вывод в EventLog SQL-запроса
    OutputDebugString( PChar(DataModule1.ADOQPost.SQL.Text) );

    DataModule1.ADOQPost.ExecSQL;
    //Показываем Поставщиков
    DataModule1.ADOQPost.Close;
    DataModule1.ADOQPost.SQL.Text:='SELECT Поставщики.Код,
Поставщики.Наименование, '
    +' Поставщики.ФИО, Поставщики.Телефон,
ГруппыПродуктов.Наименование '
    +' FROM ГруппыПродуктов INNER JOIN Поставщики ON
ГруппыПродуктов.Код = Поставщики.ГруппаПродуктов;';
    DataModule1.ADOQPost.Open;
    //Задаем ширину колонок для DG2
    With DBGrid2 do
    begin
        Columns.Items[0].Width:=30;
        Columns.Items[1].Width:=200;
        Columns.Items[2].Width:=150;
        Columns.Items[3].Width:=100;
        Columns.Items[4].Width:=150;
    end;
    Panel4.Visible:=true;
end;

procedure TForm4.DBGrid2CellClick(Column: TColumn);
begin
    currentSelectedRecordId := DBGrid2.Fields[0].AsInteger;
end;

end.
```