

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Г. М. Цибульский

подпись

« _____ » _____ 20 ____ г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02.05 «Информационные системы и технологии в административном
управлении»

Автоматизация формирования заказов поставщику для предприятия ООО ПКФ
«Карунд-ч»

Руководитель _____ ст. преподаватель каф. СИИ Р. В. Брежнев
подпись, дата

Выпускник _____ Е. А. Одеменин
подпись, дата

Нормоконтролер _____ М. А. Аникьева
подпись, дата

Красноярск 2016

СОДЕРЖАНИЕ

Введение.....	3
1 Аналитическая часть.....	5
1.1 Анализ технического задания.....	5
1.2 Характеристика предприятия.....	6
1.3 Анализ существующих решений.....	7
1.3.1 «1С: Предприятие».....	7
1.3.2 «Галактика».....	10
1.3.3 «Парус».....	13
1.3.4 «SAP».....	15
1.4 Сравнение аналогов.....	17
1.5 Выводы по главе 1.....	19
2 Проектирование и реализация системы.....	20
2.1 Проектирование системы.....	20
2.1.1 Методология Use Case.....	20
2.1.2 «Актеры».....	21
2.1.3 Варианты использования.....	23
2.1.4 Диаграмма деятельности.....	27
2.1.5 Диаграмма классов.....	28
2.2 Реализация системы.....	31
2.2.1 Формы.....	31
2.2.2 Обработки.....	34
2.2.3 Встроенный язык.....	34
2.2.4 Конструктор запросов.....	36
2.2.5 Отчеты.....	41
2.3 Выводы по главе 2.....	42
Заключение.....	43
Список использованных источников.....	44
Приложения А – В.....	46-65

ВВЕДЕНИЕ

Информационные системы и технологии широко применяются в сферах производства и торговли. Первостепенной задачей Министерства финансов Красноярского края является, обеспечение единства и создание условий для эффективного функционирования финансовой системы края и финансовых систем образований края.

В современных условиях экономики 90 % предприятий и организаций применяют информационные системы, которые помогают им в осуществлении деятельности. По данным статистики за 2014 год, на большинстве предприятий, а именно 80 %, преобладает информационная среда «1С: Предприятие», потому что является наиболее удачной из всех аналогов, предлагаемых на территории России, такие продукты как «Галактика» и «Парус» распространены намного меньше и имеют доли 15 % и 5 % соответственно. Из зарубежных аналогов можно выделить SAP и Navision, которые имеют значительные доли на информационных рынках других стран. Информационные системы тесно связаны с различными стандартами, в России это ГОСТы, а за рубежом к таковым относятся европейские стандарты.

На предприятии «Карунд-ч» успешно применяется информационная система «1С: Предприятие 8.3», которая способствует более экономически выгодному ведению деятельности, так как, позволяет вести учет сразу по всем отделам предприятия, автоматизировать множество процессов, а также создавать информативные отчеты о деятельности всех отделов.

Предполагается, что автоматизация формирования заказов поставщику на данном предприятии позволит существенно ускорить работу с документацией, сократит время на обработку данных, а также, исключит человеческий фактор при формировании заказа, что благоприятно скажется на работе предприятия.

В связи с этим была выбрана тема: автоматизация формирования заказов поставщику для предприятия ООО ПКФ «Карунд-ч».

Целью данной работы будет являться успешная разработка информационной системы для формирования заказов поставщику.

В рамках данной работы решаются следующие задачи:

- обзор существующих решений;
- проектирование системы;
- разработка системы.

1 Аналитическая часть

1.1 Анализ технического задания

Целью данной работы является проектирование и разработка программного решения для автоматизации формирования заказов поставщику для предприятия ООО ПКФ «Карунд-ч», на базе существующей информационной инфраструктуры. Данное решение будет использоваться на предприятии ООО ПКФ «Карунд-ч» находящимся в городе Красноярске для ускорения работы с документацией, сокращения времени на обработку данных и максимальному исключению человеческого фактора при формировании заказа.

Итоговый продукт должен выполнять следующие функции:

- формирование заказа;
- выбор склада или складов для подсчета остатков изделий;
- подсчет необходимых для заказа изделий;
- разделение изделий на комплектующие;
- выбор поставщика;
- создание информативных отчетов о заказе.

Также продукт должен отвечать требованиям к системе, а именно:

- система должна быть установлена на рабочих ПК всех отделов предприятия ООО ПКФ «Карунд-ч»;
- система должна иметь «поиск по складу», то есть, считать остатки деталей на выбранном складе или нескольких складах;
- во вкладке «Продукция» при создании заказа, должны присутствовать такие поля, как: номер изделия, полное название изделия, количество;
- вкладка «Комплектующие» должна отображать из каких комплектующих состоит выбранное изделие, количество остатков на складе,

необходимое количество, сколько требуется заказать у поставщика и вес данных комплектующих;

- вкладка «Сделать заказ» должна включать в себя: номер договора и дату, выбора контрагента, то есть поставщика, с которым придется работать, склад, на который придет заказ, дата поступления заказа и дата оплаты, счет на который необходимо перевести средства;

- также во вкладке «Сделать заказ» должна присутствовать вся информация о заказе, а именно, название изделия, количество, цена, НДС, общая стоимость;

- система должна при завершении заказа выводить информативный отчет о заказе, который включает в себя: реквизиты поставщика и покупателя, всю информацию о заказе и конечную стоимость.

1.2 Характеристика предприятия

Полное наименование: ООО ПКФ «Карунд-ч».

Организационно-правовая форма: Общество с ограниченной ответственностью.

Компания осуществляет самостоятельную деятельность с июня 2003 года, как фирма, сформированная из квалифицированных специалистов широкого профиля. Ориентируя свою деятельность на производстве, доработке, установке, перепродаже, торгового и холодильного оборудования.

В настоящий момент компания имеет все необходимые для своей деятельности сертификаты и лицензии. Значительный опыт работы с высокотехнологичным оборудованием, отработанная логика принятия решений, знание специфики, гибкая ценовая политика.

Основные направления деятельности ООО ПКФ «Карунд-ч»:

- комплексное оснащение торговым, технологическим, холодильным оборудованием для предприятий торговли и общепита;

- оборудование для хлебопекарен, кондитерских, молочных и мясоперерабатывающих цехов;
- оснащение объектов под «ключ»;
- системы кондиционирования;
- системы вентиляции;
- холодильные склады;
- расходные материалы, а так же запчасти для холодильного, технологического оборудования;
- разработка проектов и консультирование по вопросам подбора оборудования;
- монтаж, гарантия, сервисное обслуживание.

1.3 Анализ существующих решений

В данном разделе будут рассмотрены наиболее распространенные обучающие системы для образовательных учреждений. Также будет проведен сравнительный анализ систем относительно требований технического задания.

1.3.1 «1С: Предприятие»

«1С: Предприятие» является системой программ для автоматизации разных областей экономической деятельности. В конкретный программный продукт, входящий в систему программ «1С: Предприятие», определяются те функции и возможности, которые соответствуют назначению данного продукта. Все компоненты системы программ можно поделить на технологическую платформу и конфигурации. Технологическая платформа предполагает собой набор различных механизмов, применяемых для автоматизации экономической деятельности и не зависящих от определенного законодательства и методологии учета. Конфигурации являются фактически прикладными решениями. Каждая конфигурация направлена на автоматизацию

определенной сферы экономической деятельности и, соответственно, отвечает принятому законодательству [1].

Гибкость платформы позволяет применять «1С: Предприятие» в самых различных областях:

- автоматизация производственных и торговых предприятий, бюджетных и финансовых организаций, предприятий сферы обслуживания;
- поддержка оперативного управления предприятием;
- автоматизация организационной и хозяйственной деятельности;
- ведение бухгалтерского учета с несколькими планами счетов и произвольными измерениями учета, регламентированная отчетность;
- широкие возможности для управленческого учета и построения аналитической отчетности, поддержка многовалютного учета;
- решение задач планирования, бюджетирования и финансового анализа;
- расчет зарплаты и управление персоналом;
- другие области применения [2].

В системе «1С: Предприятие» присутствует четкое разделение на платформу и прикладное решение. Платформа представляет собой окно, в котором функционирует прикладное решение:

- платформа служит фундаментом для построения прикладных решений;
- платформа является средой их исполнения;
- платформа содержит инструментарий, необходимый для разработки, администрирования и поддержки прикладных решений.

При этом прикладное решение считается самостоятельной сущностью и имеет возможность выступать в виде отдельного программного продукта. Хотя полностью опирается на технологии платформы.

Прикладное решение не пишется в прямом смысле на языке программирования. Язык программирования применяется исключительно там, где это действительно необходимо.

В базе прикладного решения лежат метаданные. Они представляют собой структурированное декларативное его описание. Метаданные образуют

иерархию объектов, из которых формируются все детали прикладной системы и которые характеризуют все нюансы ее поведения. Практически, при работе прикладного решения, платформа интерпретирует метаданные, обеспечивая всю необходимую работоспособность.

Метаданными описываются структуры данных, состав типов, взаимосвязи между объектами, специфики их поведения и визуального представления, система разделения прав доступа, пользовательский интерфейс и т.д. В метаданных сосредоточены сведения не только о том, «что хранить в базе данных», но и о том, «зачем» хранится какая-нибудь информация, какова ее роль в системе, и как соединены между собой информационные массивы.

Внедрение языка программирования ограничено решением тех задач, которые действительно требуют алгоритмического описания, к примеру, расчета налогов, проверки правильности введенных данных.

Главной мыслью построения интерфейса считается максимальное использование информации из метаданных, также объектов манипулирования данными с тем, чтобы вся система не требовала детализированной настройки со стороны разработчика и работала по большей части автоматически.

Разработчику достаточно связать такой объект с составляющей формы либо с самой формой, и механизм интерфейса полностью возьмет на себя организацию просмотра и модификации данных. Платформа автоматом подключит расширения, учитывающие тип данных, с которыми связан элемент управления или форма.

Платформа имеет целый набор механизмов, позволяющих создавать приложения на различных языках. Начиная от разных языков интерфейса платформы, поддержки национальных дат, чисел, и завершая средствами редактирования текстов интерфейса [3].

Интерфейс платформы представлен на рисунке 1.1. Достаточно интуитивный и понятный даже неопытному пользователю. Все подсистемы вынесены в отдельные иконки, что делает работу более удобной и экономит

время, также в отдельные иконки вынесены и различные обработки, которые также упрощают поиск данных функций.

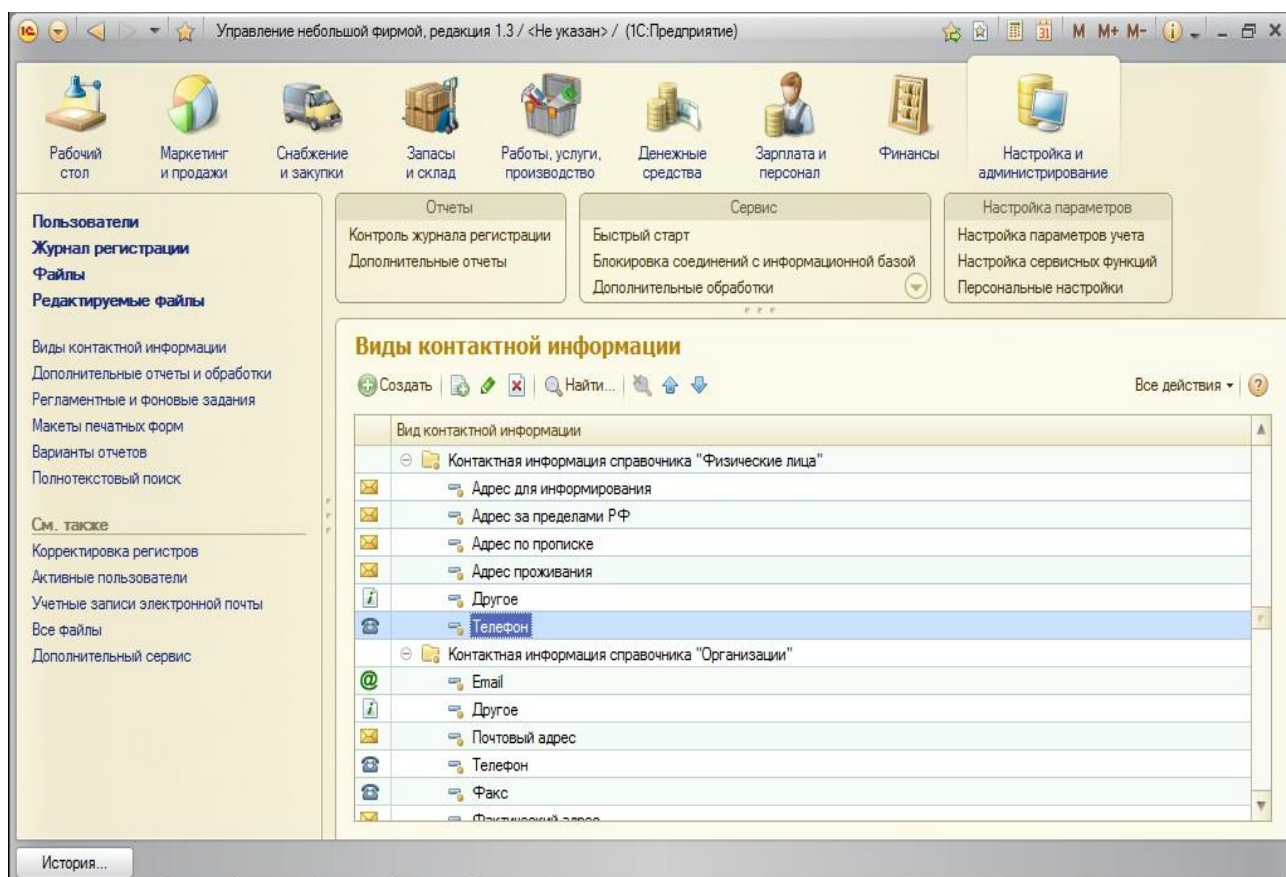


Рисунок 1.1 — Интерфейс платформы «1С: Предприятие»

1.3.2 «Галактика»

Система «Галактика» представляет собой компьютерную программу, созданную для обработки информации, образующейся в ходе экономической деятельности предприятия, направленной на получение прибыли. Системы данного рода гарантируют автоматизированное управление всеми видами ресурсов компании: материальными, финансовыми, кадровыми, позволяют осуществлять планирование и управление ходом производства, хранения, и сбыта продукции, гарантируют своевременное разумное обеспечение производства нужными материалами и комплектующими, предоставляют

достоверную оперативную информацию для принятия управленческих решений.

Программа «Галактика» построена по модульному принципу, любой модуль предназначен для автоматизации отдельной функции компании, модули объединяются в группы, контуры (в скобках указаны названия модулей):

- контур управления производством (спецификации товаров, управление заказами, управление ремонтами, учет в производстве, планирование производства, контроллинг, корпоративное планирование);

- финансовый контур (финансовый анализ, платежный календарь, управление бюджетом, планирование финансов, управление проектами);

- контур бухгалтерского учета (векселя и кредиты, фактические издержки, финансово-расчетные операции, материальные ценности, малоценные и быстро изнашиваемые предметы, ведение налоговых расчетов, налоговые регистры, основные средства, нематериальные активы, бухгалтерская отчетность, консолидация, хозяйственные операции, касса);

- контур логистики (материально-техническое обеспечение, управление договорами, управление снабжением, управление сбытом, складской учет, поставщики, получатели);

- контур управления взаимоотношениями с клиентами (клиенты, маркетинговые компании);

- контур управления персоналом (управление персоналом, зарплата).

В состав системы «Галактика» входят развитые средства для помощи решения специальных и отраслевых задач (претензионно-исковая работа, консигнация, давальческое сырье, управление строительством, гарантийное обслуживание, спецодежда, розничная торговля, автомобильный транспорт), также инструментарий для администрирования системы (настройка, рабочее место пользователя, документооборот, конфигуратор, экспорт/импорт, генератор отчетов, компилятор форм, деловая графика, компилятор интерфейсов, обмен бизнес-документами, права доступа и некие другие, созданные для настройки и администрирования системы).

Система «Галактика» используется на нескольких сотнях крупных и средних предприятий, в том числе на нескольких нижегородских: «Борский пищевой комбинат», «Оргсинтез», «Речной порт», «Машиностроительный завод», «Красный якорь», «Сибурнефтехим».

При верно налаженной организационной схеме многофункциональной эксплуатации информационной системы «Галактика» любой исполнитель выполняет определенные для него инструкции действия, получая информацию в размере, нужном и достаточном для осуществления своих должностных обязательств.

В результате работы всех пользователей информационной системы случается наполнение базы данных (БД) компании оперативной информацией о ходе выполнения определенных хозяйственных операций, имеющих отношение к различным направлениям деятельности. Обработка своевременной информации позволяет, с одной стороны, проанализировать взаимоотношения с контрагентом на базе сведений о перемещении материальных ценностей, услуг, работ и финансовых средств, а с другой стороны, расценить эффективность работы предприятия по всем направлениям хозяйственной деятельности. При всем этом обеспечивается:

- однократность ввода информации в БД и, как следствие, недоступность дублирования функций пользователей, упорядочение документооборота;
- легкость контролирования на правильность и целостность данных, персонализация действий пользователя;
- контроль над регламентом выполнения хозяйственных операций;
- стремительная перестройка комплекса, изменение эксплуатационной схемы при изменении бизнес-процесса (технологии управления) [4].

Систему невозможно корректировать, не считая незначительных поправок интерфейса. Все доработки нужно заказывать у разработчика системы, в сравнении с «1С: Предприятие» это выйдет необоснованно затратно и не приемлемо по срокам [5].

Интерфейс представлен на рисунке 1.2.

Дата	Баланс	Приход	Расход	Финансовый поток	Остаток конечный	Дефицит	План по приходу	План по расходу
06/05/2	0.00	0.00	45'648'000.00	-45'648'000.00	0.00	0.00	0.00	0.00
07/05/2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3'195'000.00
08/05/2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	165'209.13
12/05/2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2'882'059.74
13/05/2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	38'458.56
14/05/2000	0.00	0.00	0.00	0.00	0.00	0.00	243'749.00	360'303.80
15/05/2008	руб	0.00	0.00	0.00	0.00	0.00	0.00	7'320.72
21/05/2008	руб	0.00	0.00	0.00	0.00	0.00	0.00	625'843.98
22/05/2008	руб	0.00	0.00	0.00	0.00	0.00	0.00	3'312'223.02
23/05/2008	руб	0.00	0.00	0.00	0.00	0.00	0.00	303'926.70
29/05/2008	руб	0.00	0.00	0.00	0.00	0.00	0.00	12'380'522.50

V	И	Статус	Дата	Номер	Приоритет	П/Р	Контрагент	Сумма	Вал.	О	Б
		исполняемый	06/05/2008	000020	платежи первой очереди	расх	"СибирьОйлТрейд"	45'648'000.00	руб	+	с

Рисунок 1.2 — Интерфейс платформы «Галактика»

1.3.3 «Парус»

Система «Парус» создана для малых и средних хозрасчетных компаний разной отраслевой принадлежности (торговля, сфера услуг, простое производство, реклама и СМИ, общественное питание, туризм, зарубежные фирмы). Это простая, комфортная, но в то же время очень функциональная система, позволяющая автоматизировать бухгалтерский учет, основные торговые процессы и складской учет, расчет заработной платы и кадровый

учет. Может использоваться как на одном, так и на нескольких объединенных в сеть компьютерах.

Система построена по модульному принципу и представляет собой набор модулей, любой из которых предназначен для автоматизации одного из основных видов деятельности компании и может работать как в автономном режиме, так и вместе с другими модулями комплекса, образуя единое информационно-управленческое пространство масштаба компании.

Состоит из следующих модулей:

- бухгалтерия;
- реализация и склад;
- комплекс;
- учет договоров;
- комплектование;
- заработная плата;
- кадры.

Любой модуль может работать как самостоятельное приложение, хотя в полной мере преимущества модулей реализуются при применении их в качестве единого программного комплекса с совместной информационной базой.

Достоинства системы:

- простота освоения;
- широкие многофункциональные способности;
- высочайшая надежность функционирования;
- стандартные настройки на различные типы предприятий;
- сокращение издержек на автоматизацию методом выбора оптимальной комплектации с возможностью ее дальнейшего наращивания;
- возможность анализа учетных данных и информации в информационной базе [6].

В базу системы «Парус» заложены следующие базовые основы:

- архитектура клиент-сервер на базе СУБД «Oracle», схожая архитектура позволяет оптимально распределить работу между клиентскими и серверной долями системы, при этом приложение, работающее на рабочей станции, не читает записи базы данных, а посылает запросы на сервер, где они принимаются и поочередно обрабатываются. В следствии на рабочую станцию поступают лишь обработанные данные, что конструктивно сокращает информационные потоки;

- масштабируемость системы «Парус» обладает возможностью наращивания количества рабочих мест без снижения эффективности работы в целом;

- модульность — возможность автономной работы модулей системы. Система «Парус» имеет возможность приобретаться в минимальной комплектации, с возможностью добавления отдельных модулей по мере развития процесса автоматизации предприятия;

- интеграция с программными продуктами иных разработчиков — система «Парус» интегрирована с приложениями Microsoft Office, что дает возможность в приложениях Word, Excel печатать подготовленные документы, строить графики по отчетам, проводить массовый ввод информации через электронные таблицы. Система располагает средствами подготовки отчетов фактически любого вида с помощью инструментального средства для подготовки шаблонов генератора отчетов [7].

На рисунке 1.3 представлен интерфейс платформы «Парус».

1.3.4 «SAP»

Система «SAP» гарантирует полную работоспособность для комплексного управления всей административной и операционной деятельностью компании, объединяя в единую цепочку финансовый учет, процессы сбыта, производства, управления материальными потоками, планирования, взаимодействия с партнерами и поставщиками. Также система

«SAP» предоставляет инструменты для стратегического управления и анализа результатов деятельности компании.

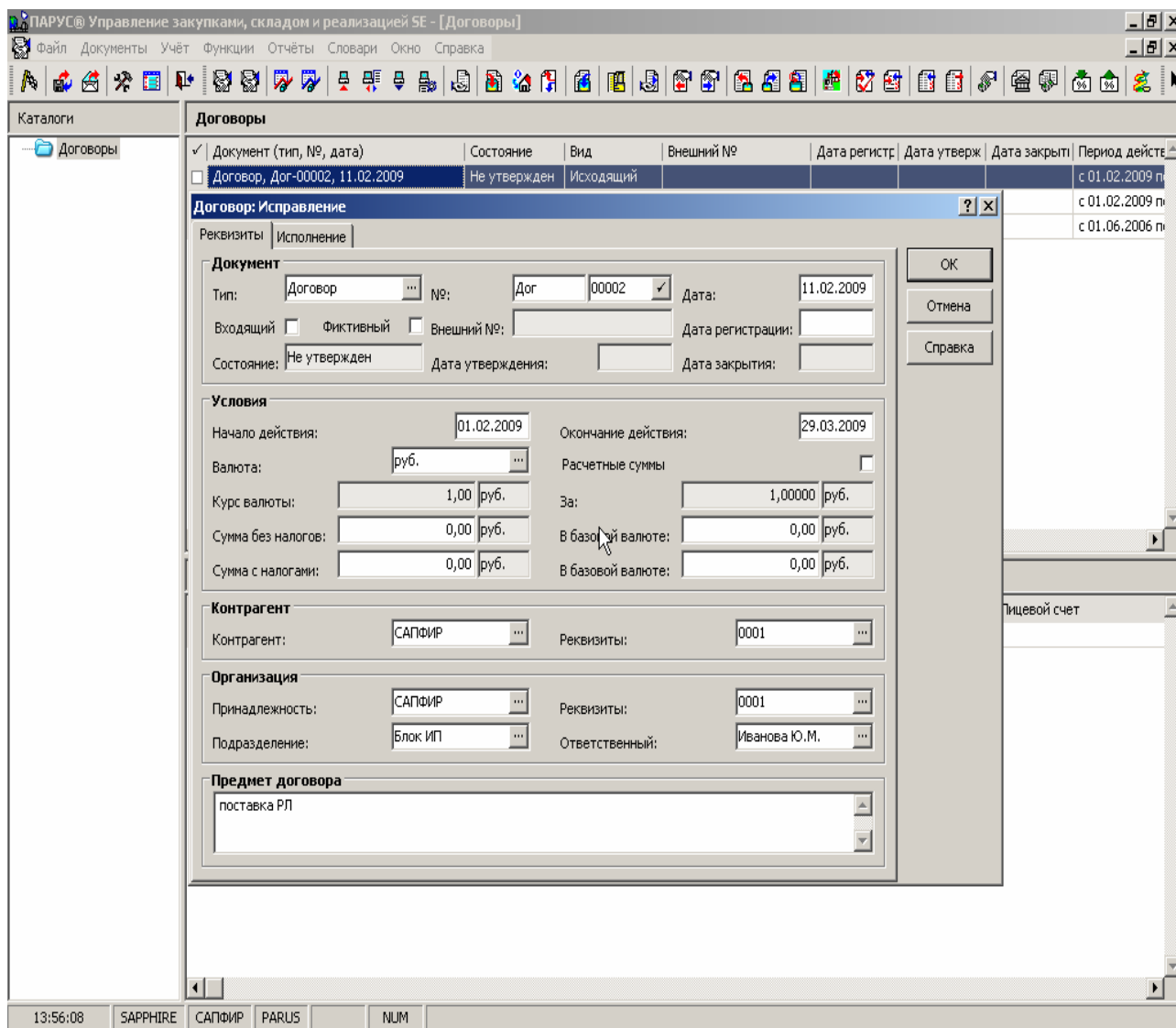


Рисунок 1.3 — Интерфейс платформы «Парус»

Решения «SAP» устраняют организационные барьеры и делают единое информационное пространство для всех подразделения и служб компании, в том числе географически удаленных друг от друга.

Говоря «SAP» обычно предполагают «SAP R/3».

«SAP R/3» — обобщённое название встроенной автоматической системы управления, которая и считается основным продаваемым продуктом разработчика «SAP».

Система «SAP R/3» состоит из комплекта прикладных модулей, которые поддерживают разные бизнес-процессы предприятия и интегрированы между собой в масштабе реального времени.

Главные особенности системы «SAP»:

- комплексная система управления, способная охватить все бизнес-процессы и сферы деятельности современного предприятия;

- интегрированная система, где все данные хранятся в единой информационной базе;

- система, работающая в реальном масштабе времени и, таким образом, способная предоставить достоверную информацию в нужной форме в нужное время [8].

Очень весомым минусом системы считается доработка системы, она возможна, но довольно трудоемкая. По практике, то, что в «1С: Предприятие» возможно подвергнуть доработке за несколько часов, в «SAP» занимает несколько дней. Обычно, в «SAP» ничего не дописывают, только настраивают.

На рисунке 1.4 представлен интерфейс платформы «SAP».

1.4 Сравнение аналогов

Рассмотрев существующие решения, можно выделить критерии, по которым можно оценить данные программные продукты. К таковым относятся: доработка системы, интерфейс, модульность, масштабируемость, цена. Рассмотрим каждый критерий отдельно.

По доработке системы явный лидер «1С: Предприятие», потому что эта среда в основном нацелена на собственноручную доработку системы под личные нужды предприятия. Если систему «1С: Предприятие» можно доработать за несколько часов, то в других продуктах на это расходуется несколько дней.

С точки зрения интерфейса опять же лидирует «1С: Предприятие», интерфейс, здесь полностью настраиваемый, все подсистемы четко разделены и не вызывают неудобства, в отличие от других программ.

Говоря о модульности, то тут нет явного лидера, можно выделить программы «Галактика» и «Парус», так как в них разработчики заранее побеспокоились обо всех составляющих предприятия, даже о самых незначительных, если в среде «1С: Предприятие» необходимо собственноручно настраивать некоторые нужные модули, то в «Галактика» и «Парус» разработчики об этом уже побеспокоились, так как доработка системы занимает очень много времени и средств.

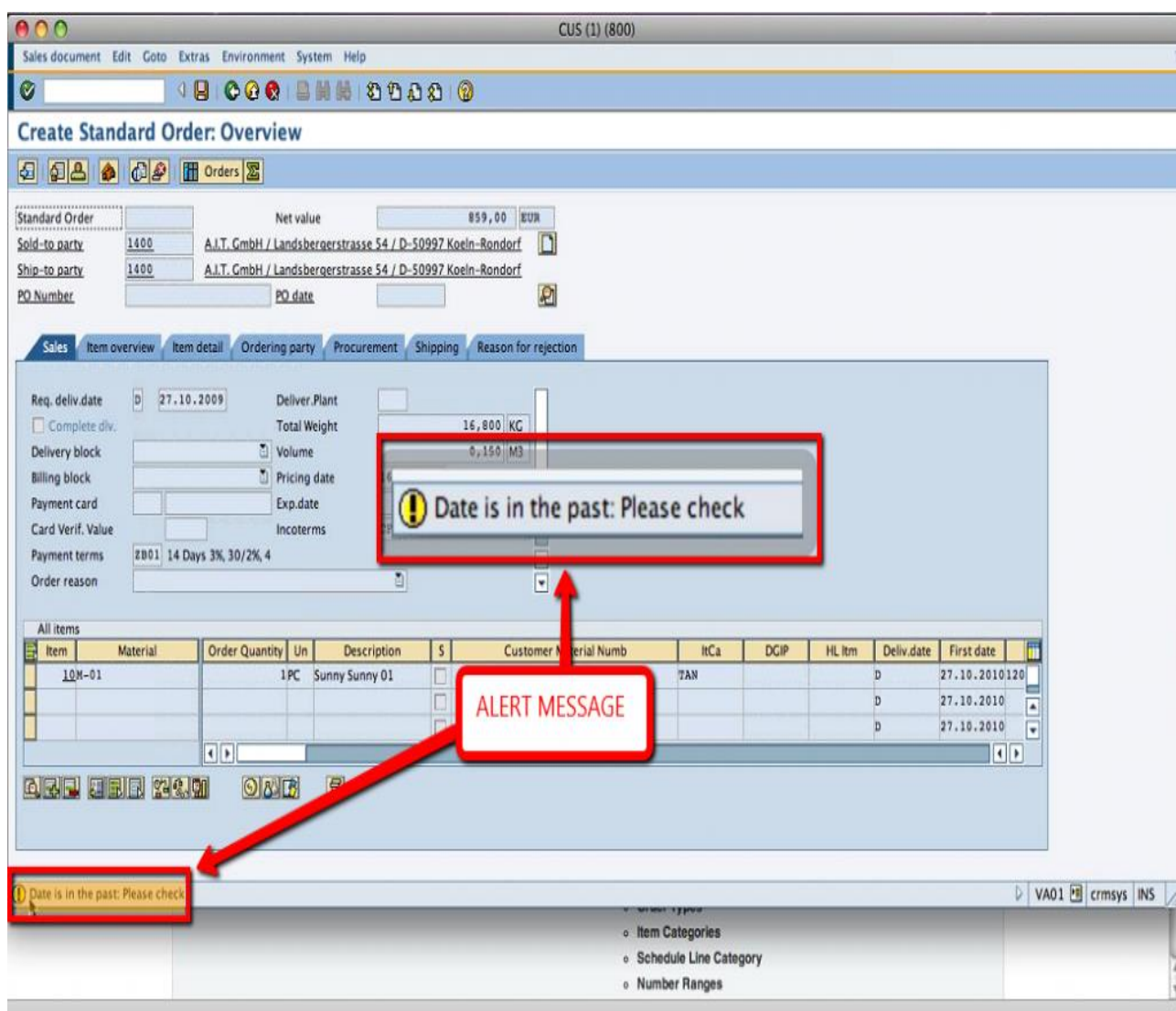


Рисунок 1.4 — Интерфейс платформы «SAP»

В плане масштабируемости все системы держатся на одном уровне, так как это одна из основ любой программы подобного характера, система должна уметь выполнять задачи, как небольшого масштаба, так и громадного с использованием огромного количества ресурсов и пользователей.

В плане стоимости продуктов, «1С: Предприятие» является несомненным лидером, так как стоимость напрямую зависит от сборки, от того сколько пользователей будет пользоваться системой и многих других факторов. «Галактика», «Парус» и «SAP» по стоимости уступают «1С: Предприятие».

В таблице 1 приведен сравнительный анализ рассмотренных программных решений по основным критериям в пяти бальной шкале оценок.

Таким образом, выбор системы «1С: Предприятие» является обоснованным и целесообразным решением.

Таблица 1 – Результаты анализа

Критерии	«1С: Предприятие»	«Галактика»	«Парус»	«SAP»
Доработка системы	5	1	1	3
Интерфейс	4	2	3	3
Модульность	4	5	5	4
Масштабируемость	5	4	5	5
Стоимость	5	3	3	1
ИТОГО:	23	15	17	16

1.5 Выводы по главе 1

В данной главе был произведен анализ технического задания, изложена характеристика предприятия, для которого разрабатывается данная система, и был произведен анализ различных программных продуктов в области управления предприятием, и на основе произведенного анализа, были сделаны выводы, и было принято обоснованное и целесообразное решение, выбрать в качестве платформы программный продукт «1С: Предприятие».

2 Проектирование и реализация системы

2.1 Проектирование системы

2.1.1 Методология Use Case

Use Case — это сценарная техника описания взаимодействия. При помощи Use Case может быть описано и пользовательское требование, и требование к взаимодействию систем, и описание взаимодействия людей и компаний в реальной жизни.

В общем случае, при помощи Use Case может описывать взаимодействие двух или большего количества участников, имеющее определенную задачу:

- покупка продукта (Покупатель-Продавец);
- отправка послания по почте (Адресант-Почтовый клиент);
- запрос страницы браузером (Браузер-Web-сервер).

В разработке ПО данную технику нередко используют для проектирования и описания взаимодействия пользователя и системы, потому название Use Case нередко принимают как синоним требования человека-пользователя к решению конкретной задачи в системе.

Исторически требования к функционированию системы описывались в виде отдельных функций. Ивар Якобсон предложил Use Case как альтернативу и дополнение описания функциональности системы. Описание требований к системе не в виде отдельных функций, а в облике описания контекста и последовательной очередности действий пользователя позволяет сформировать набор многофункциональных требований, который будет гарантировать полноту и не избыточность требований [9].

Естественно, для разработки многофункциональных требований к системе нужно писать целый набор Use Case, предусматривающих цели пользователей нескольких ролей. Данный набор позволяет обеспечить полноту требований пользователей к системе. Набор Use Case считается набором

требований более высокого уровня абстракции, нежели набор отдельных функциональных требований, и вместе с этим полностью покрывает пользовательские требования к функциональности.

Сущность этой диаграммы состоит в следующем: проектируемая система видится в виде множества сущностей либо актеров, взаимодействующих с системой при помощи так называемых вариантов использования. При этом актером (actor) или действующим лицом именуется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая иная система, которая имеет возможность быть источником воздействия на моделируемую систему так, как определит сам разработчик. Так же, вариант использования (use case) служит для описания сервисов, которые система дает актеру. Иначе, каждый вариант использования описывает некий набор действий, совершаемый системой при диалоге с актером. При всем этом ничего не говорится о том, каким образом будет реализовано взаимодействие актеров с системой.

В самом общем случае, диаграмма вариантов использования является графом особого вида, который считается графической нотацией для представления конкретных вариантов использования, актеров, возможно, некоторых интерфейсов, и отношений между этими составляющими. При этом отдельные компоненты диаграммы могут быть заключены в прямоугольник, который обозначает проектируемую систему в целом. Необходимо подчеркнуть, что отношениями этого графа могут быть лишь некие фиксированные типы взаимосвязей между актерами и вариантами использования, которые в совокупности описывают сервисы или функциональные требования к моделируемой системе [10].

2.1.2 «Актеры»

Актер представляет собой любую внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и

использует ее функциональные возможности для достижения конкретных целей или решения частных задач. При этом актеры используются для обозначения согласованного множества ролей, которые могут играть пользователи в ходе взаимодействия с проектируемой системой. Любой актер имеет возможность рассматриваться как некоторая отдельная роль относительно определенного варианта использования. Шаблонным графическим обозначением актера на диаграммах считается фигурка «человека», под которой записывается определенное имя актера.

Примерами актеров могут быть: художник, покупатель в магазине, продавец магазина, водитель, пилот самолета, директор банка, система, студент и другие сущности, имеющие отношение к концептуальной модели соответствующей предметной области.

Актеры применяются для моделирования внешних относительно к проектируемой системе сущностей, которые ведут взаимодействие с системой и используют ее в виде отдельных пользователей. В виде актеров могут выступать иные системы, подсистемы проектируемой системы либо отдельные классы. Принципиально осознавать, что любой актер характеризует некое согласованное множество ролей, в каких могут выступать пользователи этой системы в ходе взаимодействия с ней. В любой момент времени с системой ведет взаимодействие вполне определенный пользователь, при этом он играет или выступает в одной из таких ролей. Наиболее наглядный пример актера, конкретный пользователь системы со своими собственными параметрами аутентификации [11].

В проектируемой системе актерами будут выступать 2 субъекта: «Пользователь» и «Система».

На рисунке 2.1 представлено, как это выглядит.

Не обязательно показывать в диаграмме вариантов использования, что актеры взаимодействуют между собой, так как это уже предполагает сама диаграмма. Поэтому часто диаграммы создают для разных актеров.

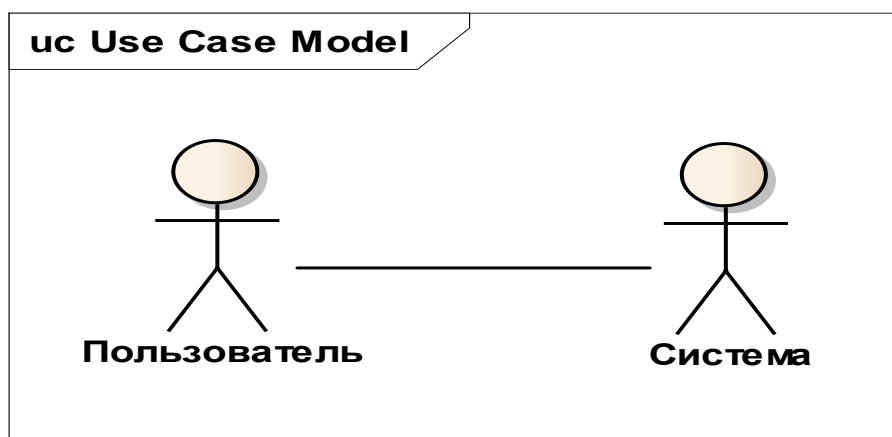


Рисунок 2.1 — Актеры в данной проектируемой системе

2.1.3 Варианты использования

Варианты использования используются для спецификации единых особенностей поведения системы либо любой другой сущности предметной области в отсутствие рассмотрения внутренней структуры данной сущности. Любой вариант использования описывает очередность действий, которые обязаны быть выполнены проектируемой системой при взаимодействии ее с соответствующим актером. Диаграмма вариантов использования имеет возможность дополняться пояснительным текстом, который раскрывает значение или семантику составляющих ее компонентов. Такой пояснительный текст получил название примечания или сценария. Отдельный вариант использования классифицируется на диаграмме эллипсом, внутри которого содержится его краткое название или имя в форме глагола с пояснительными словами.

Задача варианта использования состоит в том, чтобы определить завершенный аспект или фрагмент поведения некой сущности в отсутствие раскрытия внутренней структуры данной сущности. В виде такой сущности может выступать начальная система или любой другой элемент модели, который владеет собственным поведением, аналогично подсистеме или классу в модели системы. Любой вариант использования подходит отдельному сервису, который предоставляет моделируемую сущность или систему по

запросу пользователя (актера), т. е. описывает метод применения этой сущности. Сервис, который инициализируется по запросу пользователя, представляет собой завершенную последовательность действий. Из этого можно сделать вывод, что после того как система закончит обработку запроса пользователя, она обязана вернуться в начальное состояние, в котором готова к выполнению последующих запросов.

С системно-аналитической позиции варианты использования могут применяться как для спецификации внешних требований к проектируемой системе, но и для спецификации многофункционального поведения уже имеющейся системы. Помимо этого, варианты использования неявно устанавливают требования, характеризующие, как пользователи обязаны взаимодействовать с системой, чтобы иметь возможность корректно работать с предоставляемыми данной системой сервисами. Использование вариантов использования на всех уровнях диаграммы позволяет не только лишь достигнуть необходимого значения унификации обозначений для представления функциональности подсистем и системы в общем, но и считается мощным средством поочередного уточнения требований к проектируемой системе на базе полу уровневого спуска от пакетов системы к операциям классов. С другой стороны, модификация отдельных операций класса может оказать обратное воздействие на уточнение сервиса соответственного варианта использования, т. е. воплотить результат обратной связи с целью уточнения спецификаций или требований на уровне пакетов системы.

Образцами вариантов использования могут считаться следующие действия: оформление заказа на покупку, получение дополнительных сведений о клиенте, отображение информации на мониторе и другие действия.

На рисунке 2.2 рассмотрим варианты использования с точки зрения актера «Пользователь».

Как видно из диаграммы, вариантами использования выступают прецеденты «Выбор склада», «Выбор поставщика», «Формирование заказа», «Работа с базой данных», «Заполнение реквизитов».

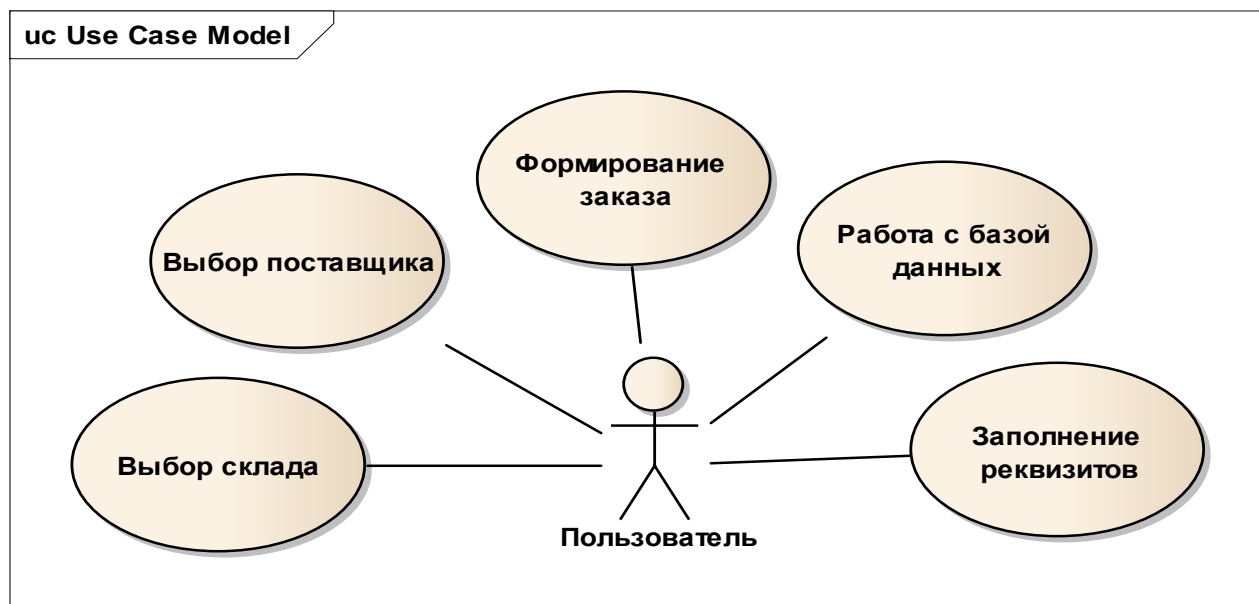


Рисунок 2.2 — Варианты использования актера «Пользователь»

К тому же некоторые прецеденты, возможно, расписать на наиболее подробные варианты использования при помощи отношения включения (include). Отношение включения между двумя вариантами использования показывает, что некое данное поведение для одного варианта использования включается в виде составного компонента в последовательность поведения другого варианта использования. Это отношение считается направленным бинарным отношением в том смысле, что пара экземпляров вариантов использования всегда упорядочена в отношении включения.

На рисунке 2.3, рассмотрим варианты использования актера «Пользователь» с отношениями включения.

Также рассмотрим варианты использования актера «Система».

Диаграмма представлена на рисунке 2.4.

Из диаграммы видно, что прецедентами являются «Разделение изделия на комплектующие», «Расчет веса комплектующих», «Расчет стоимости заказа»,

«Работа со складами», «Расчет необходимого количества изделий для заказа», «Отчет».

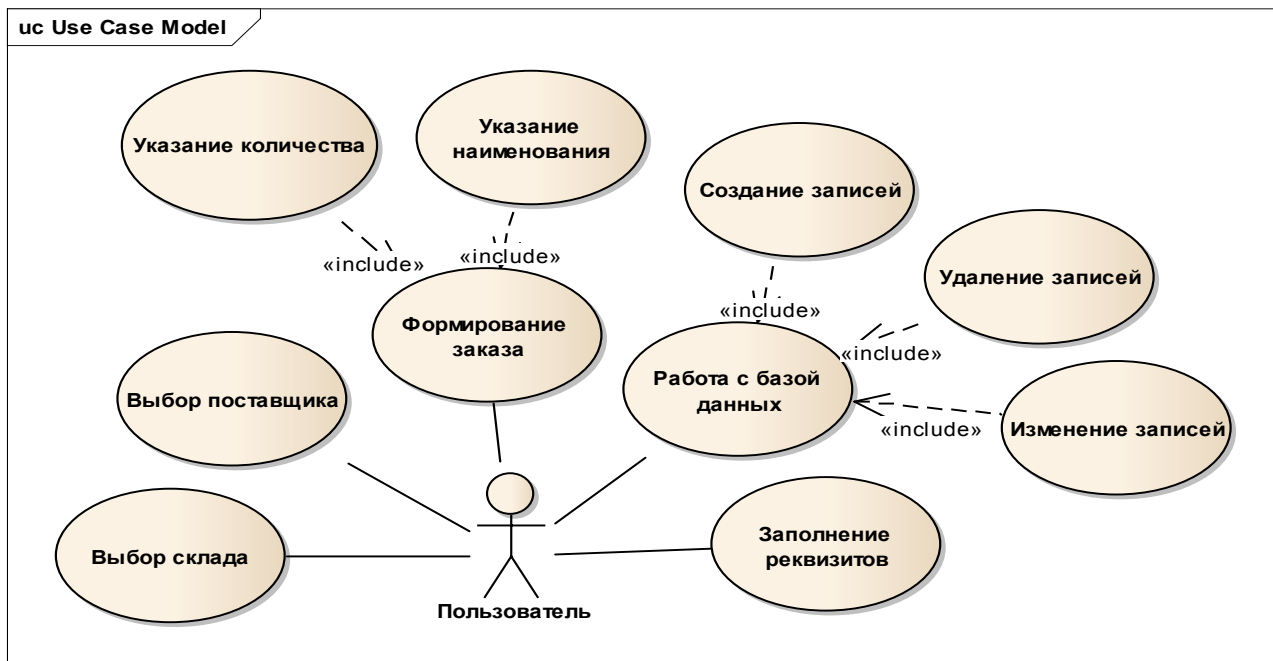


Рисунок 2.3 — Варианты использования актера «Пользователь» с отношениями включения.

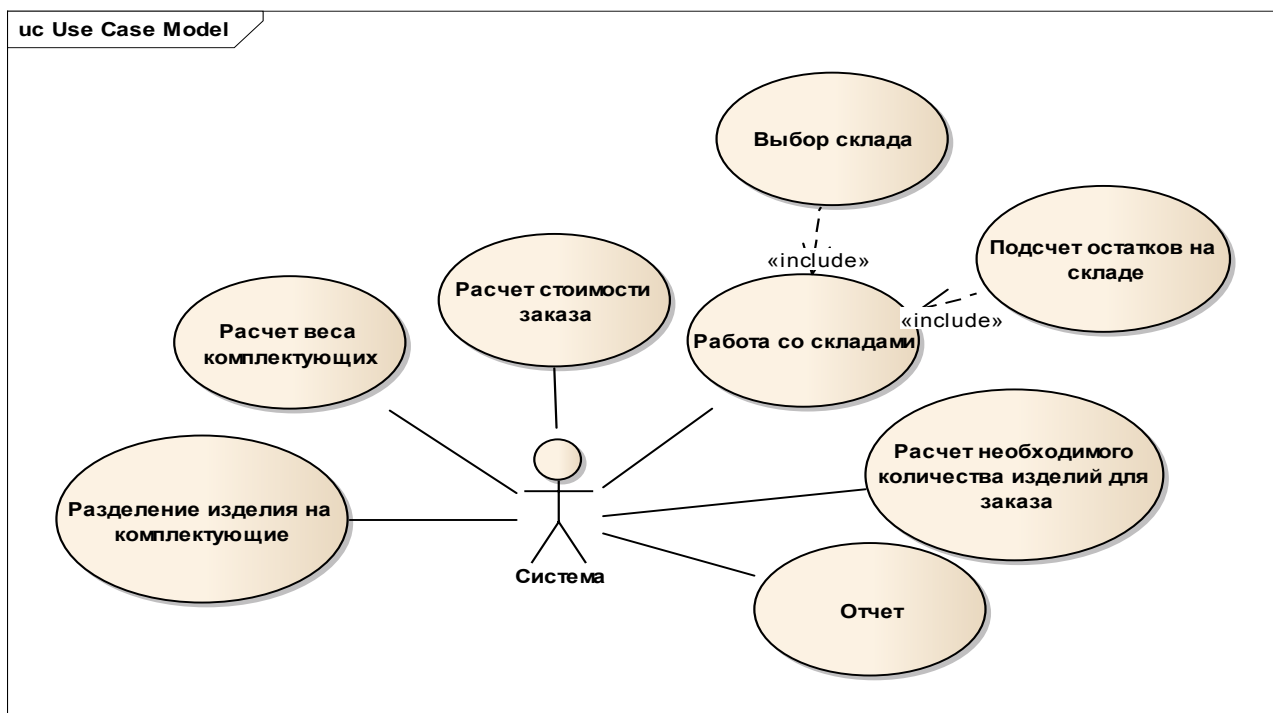


Рисунок 2.4 — Варианты использования актера «Система»

2.1.4 Диаграмма деятельности

При моделировании поведения проектируемой или анализируемой системы появляется потребность не только лишь представить процесс изменения ее состояний, но и детализировать особенности алгоритмической и логической реализации исполняемых системой операций. Обычно для данной цели применялись блок-схемы либо структурные схемы алгоритмов. Каждая такая схема акцентирует внимание на очередности выполнения определенных действий или элементарных операций, которые в совокупности приводят к получению желаемого результата.

Алгоритмические и логические операции, требующие исполнения в конкретной последовательности, окружают нас постоянно. Естественно, мы не всегда задумываемся про то, что аналогичные операции относятся к столь научным категориям. К примеру, чтобы позвонить по телефону, нам заранее нужно снять трубку или включить его. Для приготовления кофе или заваривания чая нужно сначала вскипятить воду. Чтобы выполнить ремонт двигателя автомобиля, потребуется осуществить целый ряд нетривиальных операций, таких как разборка силового агрегата, снятие генератора и некоторых других.

Для моделирования процесса выполнения операций используются так называемые диаграммы деятельности. Используемая в них графическая нотация во многом похожа на нотацию диаграммы состояний, так как на диаграммах деятельности также присутствуют обозначения состояний и переходов. Различие заключается в семантике состояний, которые используются для представления не деятельности, а действий, и в отсутствии на переходах сигнатуры событий. Любое состояние на диаграмме деятельности соответствует выполнению некоторой элементарной операции, а переход в последующее состояние срабатывает лишь при окончании этой операции в прошлом состоянии. Графически диаграмма деятельности видится в форме

графа деятельности, вершинами которого являются состояния действия, а дугами, переходы от одного состояния действия к другому.

Можно сделать вывод, что диаграммы деятельности можно считать частным случаем диаграмм состояний. Именно они позволяют воплотить особенности процедурного и синхронного управления, обусловленного окончанием внутренних деятельностей и действий. Мета модель предоставляет для этого необходимые термины и семантику. Главным направлением использования диаграмм деятельности считается визуализация особенностей реализации операций классов, когда нужно представить алгоритмы их выполнения. При всем этом каждое состояние может считаться выполнением операции некоторого класса либо ее части, позволяя применять диаграммы деятельности для описания реакций на внутренние события системы [12].

В контексте деятельность представляет собой некоторую совокупность отдельных вычислений, выполняемых автоматом. При этом отдельные простые вычисления могут приводить к некоторому результату или действию. На диаграмме деятельности отображается логика или последовательность перехода от одной деятельности к другой, при этом внимание фокусируется на результате деятельности. Сам же результат может привести к изменению состояния системы или возвращению некоторого значения [13].

Диаграмма деятельности данного проекта изображена на рисунке 2.5.

2.1.5 Диаграмма классов

Диаграммы классов считаются важнейшим звеном методологии объектно-ориентированного анализа и проектирования.

Диаграмма классов указывает классы и их отношения, тем самым представляя логический аспект проекта. Отдельная диаграмма классов представляет определенный ракурс структуры классов.

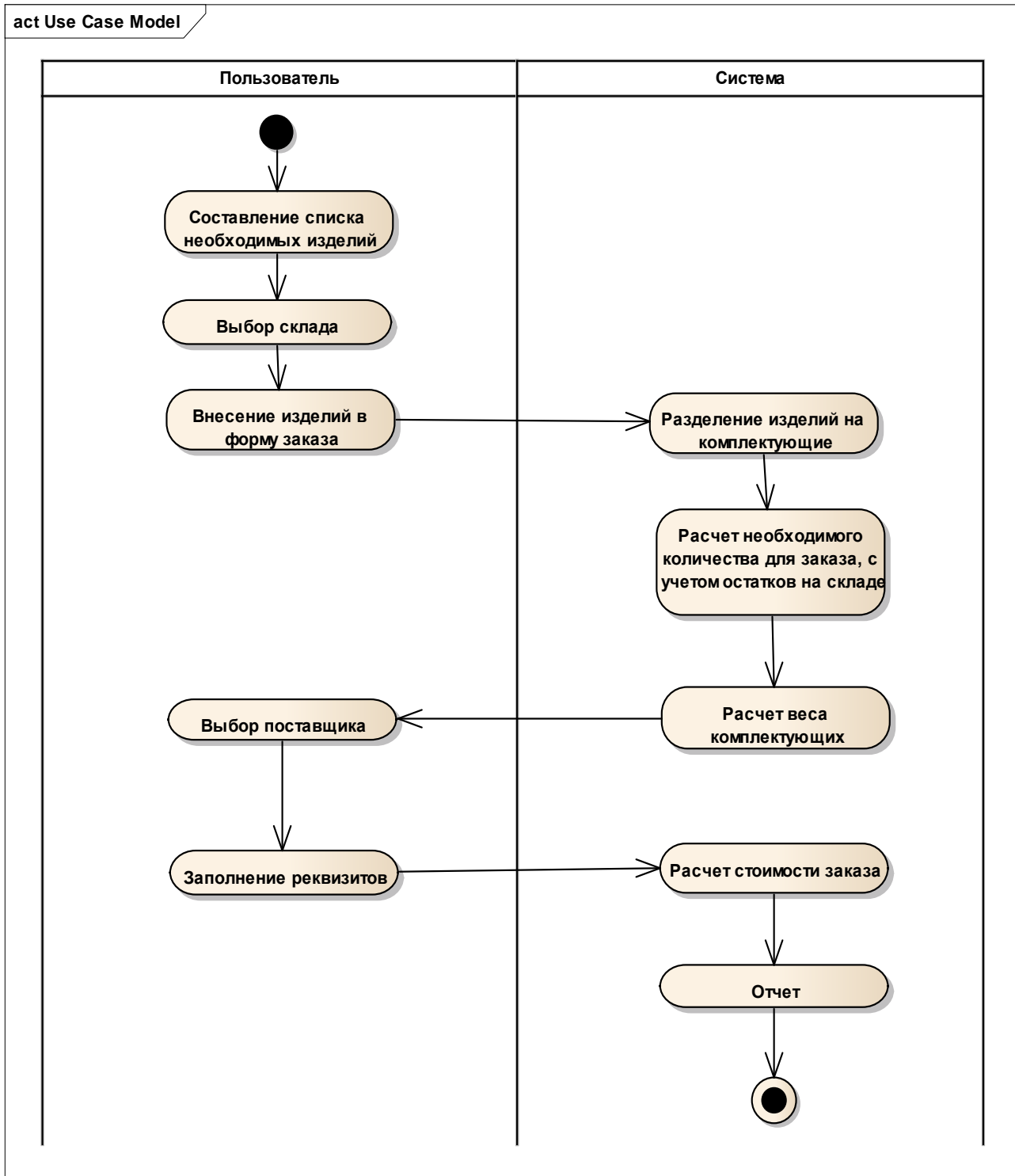


Рисунок 2.5 — Диаграмма деятельности проектируемой системы

На стадии анализа диаграммы классов используются, чтобы выделить совместные роли и обязанности сущностей, обеспечивающих требуемое поведение системы.

На стадии проектирования диаграммы классов используются, чтобы передать структуру классов, формирующую архитектуру системы.

Любой класс должен иметь имя, если имя слишком длинное, его можно сократить либо увеличить сам значок на диаграмме. Имя каждого класса должно быть уникально в содержащем его проекте.

Диаграмма классов характеризует разновидности объектов системы и различного рода статические взаимосвязи, которые существуют между ними. Имеется два главных вида статических взаимосвязей:

- ассоциации (к примеру, менеджер имеет возможность вести несколько проектов);

- подтипы (работник является разновидностью личности).

На диаграммах классов изображаются также атрибуты классов, операции и ограничения, которые накладываются на связи между объектами [14].

Класс может не иметь экземпляров или объектов. В таком случае он называется абстрактным классом, а для обозначения его имени используется курсив.

Атрибуты класса или свойства записываются во второй сверху секции прямоугольника класса.

Операции или методы класса записываются в третьей сверху секции прямоугольника. Операция представляет собой некоторый сервис, предоставляемый любым экземпляром класса по конкретному требованию. Совокупность операций характеризует функциональный нюанс поведения класса. Запись операций класса также стандартизована и подчиняется определенным синтаксическим правилам. При всем этом любой операции класса соответствует отдельная строка, которая состоит из квантора видимости операции, имени операции, выражения типа возвращаемого операцией значения и, возможно, строка-свойство данной операции [15].

На рисунке 2.6, представлена диаграмма классов к разрабатываемой системе формирования заказов поставщику.

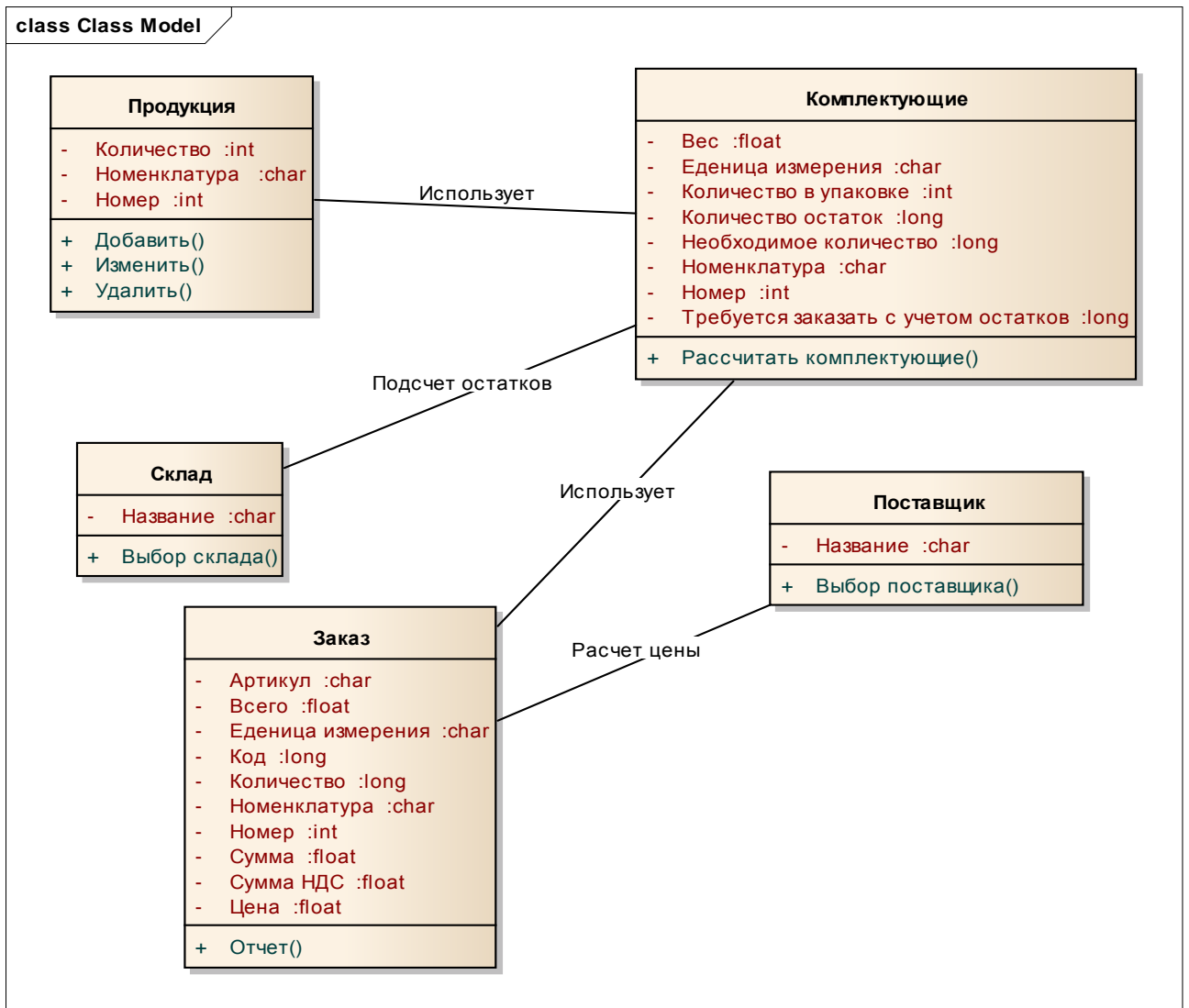


Рисунок 2.6 — Диаграмма классов проектируемой системы

2.2 Реализация системы

2.2.1 Формы

Форма — это способ представления объекта пользователю. Обычно форма представляет собой ряд полей, которые требуется заполнить и набор кнопок (команд, меню) для управления. Форма есть практически у всех объектов.

В толстом клиенте используются «обычные» формы. Это означает, что пользователь рисует мышкой форму, также как это происходит в Visual Studio.

В тонком клиенте и веб клиенте употребляются управляемые формы. Это означает, что их размер, форму и расположение их полей невозможно изменить мышкой. Они создаются автоматом на основании настроек клиента. Подавляющее большинство форм типовых конфигураций имеют свое, типовое представление, привычное для пользователей.

Форма списка. В «1С: Предприятие» есть множество списков, когда употребляется выражение «справочник номенклатуры», то подразумевается «список элементов номенклатуры». Когда говорится «документы банка», то имеется в виду «список документов ПКО и РКО». Для отображения списков есть формы списков. Это всегда табличная часть на всю форму и над ней командная панель с кнопками «Добавить», «Скопировать», изображаемых в виде картинок.

Форма элемента и документа. Когда с помощью списка пользователь определяется, с каким определенным документом он готовится работать, открывается форма документа или элемента справочника. Она, как правило, состоит из шапки (списка обязательных полей) и закладок с таблицами. В верхней части находится также панель с кнопками, на ней обычно отображаются стандартные функции «Действия», «Цены и валюта», «Операция».

Форма выбора. Очень схожа с формой списка. Часто форма списка используется в качестве формы выбора. Она используется только тогда, когда при заполнении определенного документа или элемента справочника пользователь нажимает кнопку выбора и открывается форма для выбора необходимого значения.

Форма обработки. Обычно имеет две командных панели, которые расположены сверху и снизу, кнопки запуска и закрытия формы имеют стандартное расположение. У отчетов и обработок формы списка нет [16].

В данной системе будет использоваться форма обработки, и будет она выглядеть следующим образом.

Макет изображен на рисунке 2.7.

Как описывалось выше, имеет две командные панели: панель «Продукция», где будет указываться изделие и ее количество, которое будет необходимо заказать. А также панель «Комплектующие», где будет размещено наименование комплектующего, единица измерения, количество в упаковке, количественный остаток на выбранном складе, необходимое количество, и количество с учетом остатка на складе, а также вес комплектующих.

Обработка ФормированиеЗаказовПоставщику: Форма

Формирование заказов поставщику

Действия

Продукция

N	Номенклатура	Количество
<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>

Комплектующие

Склад:

Расчитать комплектующие

N	Номенклатура	Единица измерения	Количество в упаковке	Количество остаток	Необходимое количество	Требуется заказать(с уч. упак.)	Вес
<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>

Сделать заказ Закрыть

Диалог Модуль Реквизиты

Рисунок 2.7 — Макет разрабатываемой системы

2.2.2 Обработки

Для реализации в «1С: Предприятие» инструментов для дальнейшего облегчения работы пользователя, а также, администратора базы данных присутствует специальный объект конфигурации обработки. Обработки могут использоваться как часть конфигурации или существовать как отдельные файлы, тогда такая обработка именуется внешней.

Обработки дают возможность пользователям создавать собственные инструменты, как для программирования, так и для доскональной настройки или управления. Благодаря этому в интернете присутствуют сотни типовых обработок, и тысячи любительских обработок.

Обработка является инструментом, созданным пользователем. Она ничего не умеет делать самостоятельно и не сохраняет данные в базу данных, в отличие, например от документа, который имеет возможность, сохраняться в базу данных и самостоятельно умеет записываться и проводиться, без дополнительного программирования. Функции обработки целиком зависят от программиста, который ее написал [17].

2.2.3 Встроенный язык

Встроенный язык является достаточно важной частью технологической платформы «1С: Предприятие», так как дает возможность разработчику описывать собственные алгоритмы функционирования прикладного решения.

Встроенный язык очень похож на другие языки, такие как Pascal, JavaScript, Basic, что, несомненно, облегчает его изучение начинающими разработчиками. Но стоит отметить, что он не является прямым аналогом какого-либо из перечисленных выше языков программирования.

Вот лишь некоторые, наиболее значимые и отличаемые особенности встроенного языка:

- перед исполнением модули, содержащие текст на встроенном языке, преобразуются во внутренний программный код;
- кэширование скомпилированных модулей в памяти;
- тип переменной определяется типом значения, которое она содержит, и может изменяться в процессе работы;
- разработчик может использовать как встроенные в платформу объекты, так и объекты, созданные системой в результате визуального конструирования прикладного решения [18].

Создание новых классов во встроенном языке «1С: Предприятие» запрещено. Так как платформа «1С: Предприятие» изначально нацелена на достижение задач учета, именно поэтому состав классов заранее предопределен:

- документы;
- справочники;
- регистры бухгалтерии;
- регистры накопления.

На основании типовых классов в конфигураторе можно создать любое количество подклассов имеющих свои наборы. С помощью модулей менеджеров можно немного расширить функционал подкласса [19].

После создания нужных форм обработки, для должного функционирования, необходимо запрограммировать кнопку «Расчитать комплектующие» для того, чтобы система могла разбивать необходимые изделия на комплектующие, проверять количественный остаток на складе, и на основе этого остатка рассчитывать, сколько комплектующих необходимо заказать у поставщика, а также рассчитывать вес каждого комплектующего и совокупный вес всего заказа. А также необходимо запрограммировать кнопку «Сделать заказ» для дальнейшего перехода к следующим формам: выбору поставщика, заполнению реквизитам, и отчету.

2.2.4 Конструктор запросов

Конструктор запросов - это один из инструментов разработки. Он позволяет составить текст запроса на языке запросов только визуальными средствами. Эффектом работы конструктора считается синтаксически правильный текст необходимого запроса. Тем самым, пользователь может создать полностью правильный запрос, даже не владея синтаксисом языка запросов, необходимые синтаксические конструкции конструктор сгенерирует автоматом. Готовый текст запроса может быть сразу же вставлен в текст модуля либо скопирован в буфер обмена. Помимо этого конструктор запросов позволяет изменять уже имеющийся в программе текст запроса. Для этого необходимо установить курсор внутри существующего текста запроса и вызвать конструктор. Имеющийся текст запроса будет проанализирован и показан в конструкторе в виде соответствующих выбранных полей базы данных и набора заданных связей, группировок, условий. На рисунке 2.8, можно увидеть, как выглядит конструктор запросов «1С: Предприятие» [20].

Сам запрос пишется практически автоматически, за исключением некоторых моментов, которые необходимо дописывать исключительно вручную, потому что машина либо не понимает, что хочет сделать разработчик, либо в конструкторе попросту нет таких функций.

Создаем в конструкторе запросов необходимый нам программный код для функционирования кнопок «Рассчитать комплектующие» (приложение А), и «Сделать заказ» (приложение Б).

Далее необходимо полученный программный код поместить в модуль объекта, чтобы все начало функционировать должным образом.

На рисунке 2.9, изображен модуль объекта.

После разработки программных кодов и дальнейшего их помещения в модуль объекта, обработка преобразилась и по сравнению с макетом, имеет следующий вид, и выполняет все необходимые функции, заявленные в техническом задании.

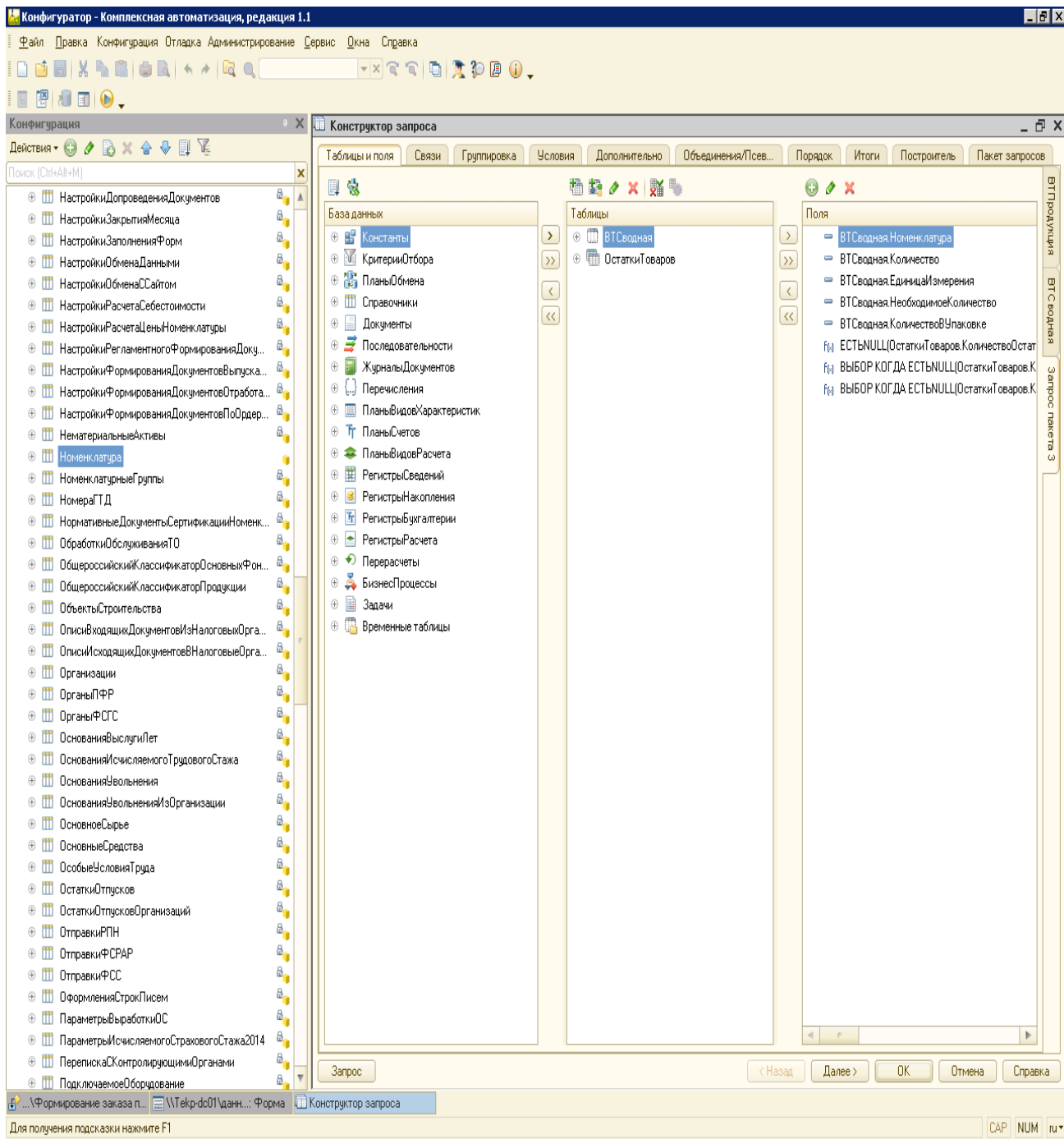


Рисунок 2.8 — Конструктор запросов 1С

После разработки программных кодов и дальнейшего их помещения в модуль объекта, обработка преобразилась и по сравнению с макетом, имеет следующий вид, и выполняет все необходимые функции, заявленные в техническом задании.

На рисунке 2.10, изображена обработка с действующим программным кодом.

На рисунке 2.11, представлена вкладка «Сделать заказ», на которой изображено окно, где указана уже конечная информация по сформированному заказу.

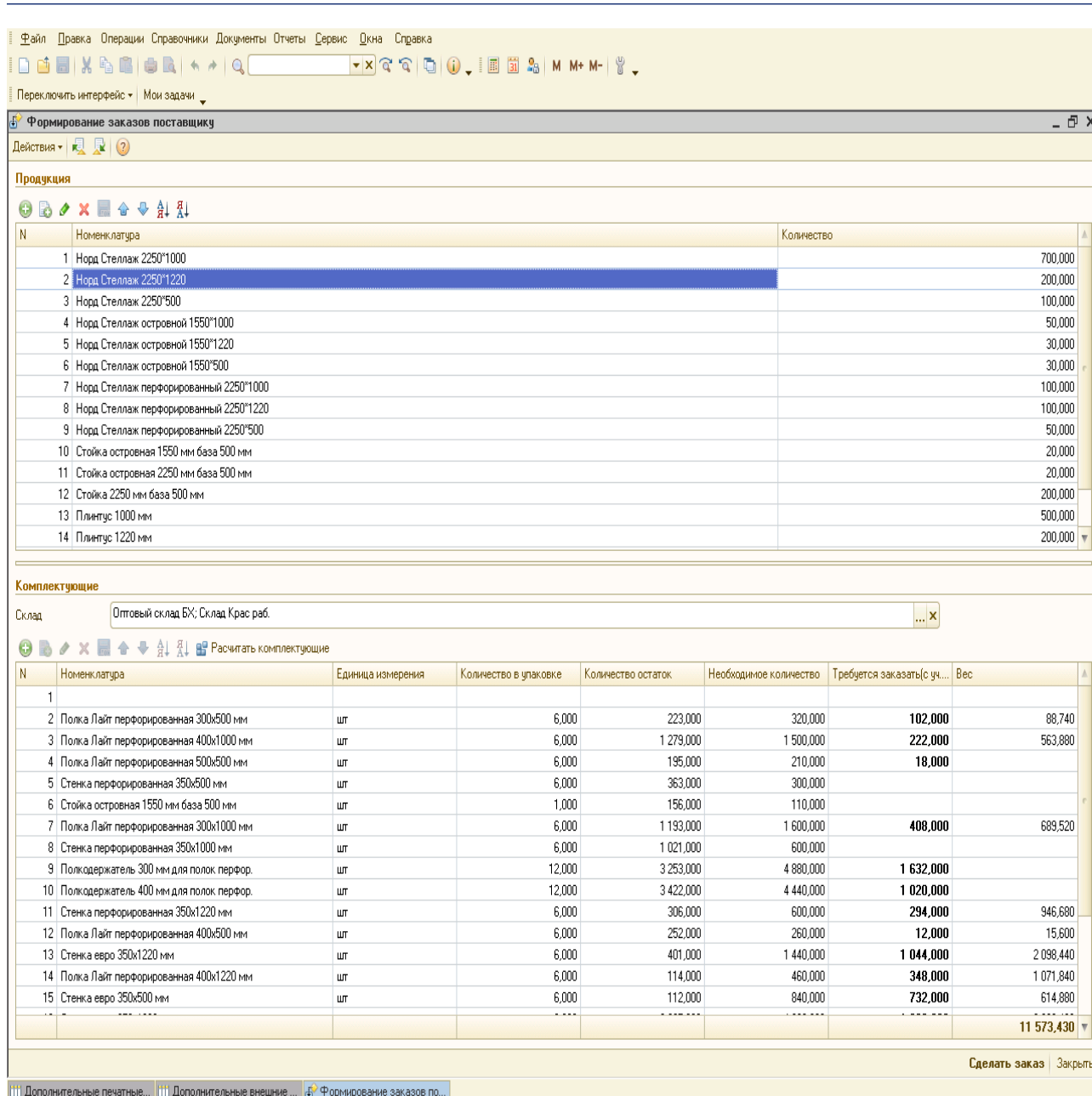


Рисунок 2.10 — Обработка с действующим программным кодом

Во вкладке «Сделать заказ» располагаются следующие элементы формирования заказа: номер, код изделия, наименование поставщика, наименование комплектующего, количество, цена, сумма, всего.

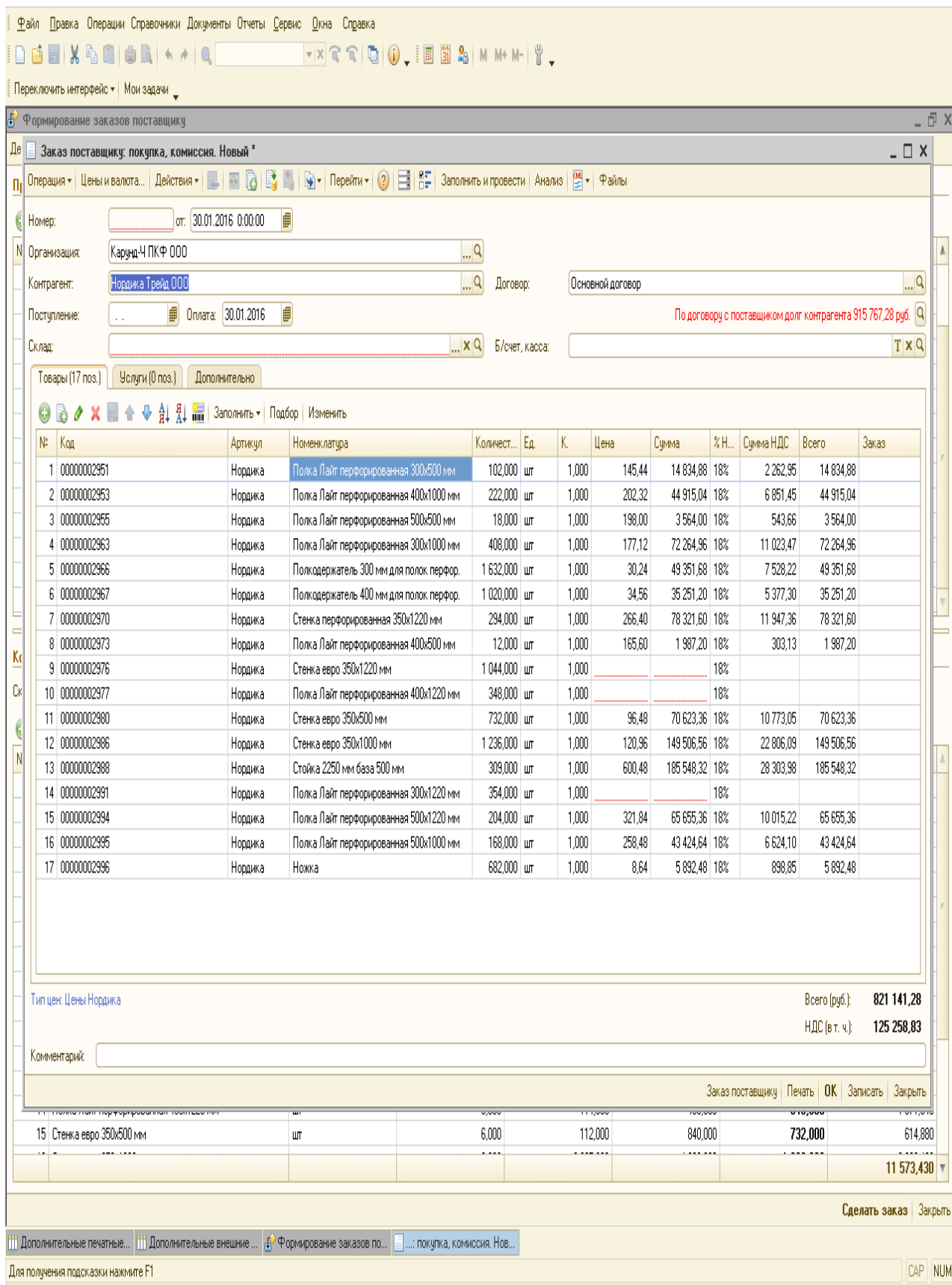


Рисунок 2.11 — Вкладка «Сделать заказ»

2.2.5 Отчеты

Отчетом можно считать специальный объект метаданных, необходимый для формирования удобного для пользователя представления данных из таблиц базы данных. Весь учет в системе, ведется для единственной цели, просмотра достигнутых результатов в ходе произведенной работы. Результатами отчета будут являться конечные цифры, посчитанные по различным регистрам.

Таким образом, при работе с отчетами есть основные базовые принципы:

- результатом отчета являются цифры в разрезе;
- справочники можно сгруппировать по строкам, по колонкам, вложено;
- отчет создается на определенную дату или за конкретный период;
- существуют отборы, позволяющие смотреть результаты только по необходимым справочникам.

Отчеты в толстом клиенте, как правило, выводят в отдельное меню. В тонком клиенте специальной закладки «Отчеты» нет, и поэтому, они разбросаны по всем закладкам.

Для администратора присутствует стандартное меню, которое содержит все документы существующей конфигурации, в которой непосредственно ведется работа. Это меню «Операции и отчеты», через которое можно открыть любой отчет [21].

На рисунке 2.12, показано, как выглядит конечный отчет разрабатываемой системы.

Он представляет собой форму заказа, в которой указаны все реквизиты, как поставщика, так и покупателя, саму таблицу, где указаны поставщик, у которого совершается заказ, наименование комплектующего, количество изделий, цена, и общая сумма с учетом НДС.

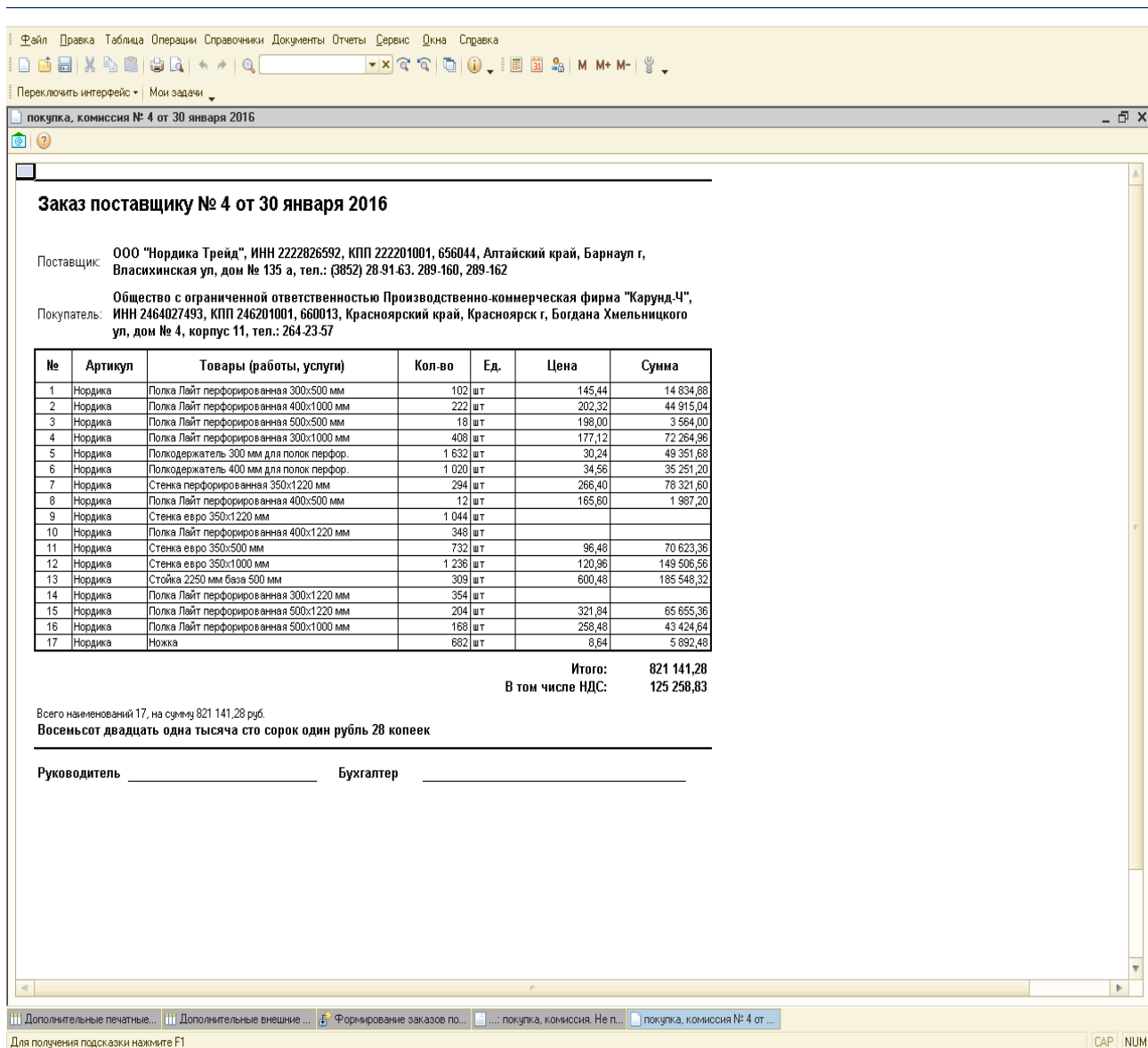


Рисунок 2.12 — Конечный отчет

2.3 Выводы по главе 2

В данной главе велась работа по проектированию и реализации разрабатываемой системы, были разработаны диаграммы, с помощью которых были наглядно описаны процессы, происходящие в системе, на основе этих диаграмм велась непосредственно реализация системы. В итоге проделанной работы была разработана и реализована система формирования заказов поставщику для предприятия ООО ПКФ «Карунд-ч».

ЗАКЛЮЧЕНИЕ

В процессе выполнения данной работы были сформированы требования к системе, функциональные возможности, рассмотрены существующие аналоги и изучены средства для разработки.

Также были разработаны диаграммы использования и деятельности, показывающие наглядно, какими функциональными возможностями обладает система, и с помощью прикладных средств система была реализована.

Поставленные цели и задачи были выполнены, данная система успешно реализована и успешно применяется на предприятии ООО ПКФ «Карунд-ч».

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Общие сведения о «1С: Предприятие» [Электронный ресурс]: // Режим доступа: <http://www.bestreferat.ru/referat-176426.html>
- 2) Области применения «1С: Предприятие» [Электронный ресурс]: // Режим доступа: <http://v8.1c.ru/overview/>
- 3) Обзор архитектуры платформы «1С: Предприятие» [Электронный ресурс] // Режим доступа: http://v8.1c.ru/overview/Term_000000877.htm
- 4) Система «Галактика» [Электронный ресурс]: // Режим доступа: <http://economy-ru.com/sistemyi-ekonomike-informatsionnyie/sistema-galaktika-25319.html>
- 5) Система «Галактика» [Электронный ресурс]: // Режим доступа: <http://programmist1s.ru/est-li-analog-1s/>
- 6) Система «Парус» [Электронный ресурс]: // Режим доступа: <http://www.parus.com/solutions/middle/products/parus7/>
- 7) Назначение и возможности ИС «Парус» [Электронный ресурс]: // Режим доступа: http://coolreferat.com/Назначение_и_возможности_информационной_системы_Парус
- 8) Система «SAP» [Электронный ресурс]: // Режим доступа: <http://www.smartek.ru/solutions/ems/sap.aspx>
- 9) Use Case [Электронный ресурс]: // Режим доступа: <http://school.system-analysis.ru/use-case/>
- 10) Use Case [Электронный ресурс]: // Режим доступа: http://www.nnre.ru/kompyutery_i_internet/samouchitel_uml/p4.php#fb2note_35
- 11) Варианты использования [Электронный ресурс]: // Режим доступа: <https://habrahabr.ru/post/47940/>
- 12) Диаграмма деятельности [Электронный ресурс]: // Режим доступа: http://www.telenir.net/uchebniki/samouchitel_uml/p7.php

13) Диаграмма деятельности [Электронный ресурс]: // Режим доступа:
<http://www.planerka.info/item/Diagrammy-deyatelnosti-UML>

14) Диаграмма классов [Электронный ресурс]: // Режим доступа:
<http://itteach.ru/rational-rose/diagrammi-klassov/vse-stranitsi>

15) Диаграмма классов [Электронный ресурс]: // Режим доступа:
http://www.info-system.ru/designing/methodology/uml/theory/class_diagram_theory.html

16) Формы 1С [Электронный ресурс]: // Режим доступа:
<http://howknow1c.ru/programmirovanie-1c/formy-1s.html>

17) Обработки 1С [Электронный ресурс]: // Режим доступа:
<http://howknow1c.ru/programmirovanie-1c/obrabotki-1s.html>

18) Встроенный язык 1С [Электронный ресурс]: // Режим доступа:
http://v8.1c.ru/overview/Term_000000020.htm

19) Встроенный язык 1С [Электронный ресурс]: // Режим доступа:
<http://programmist1s.ru/opisaniya-yazyika-programmirovaniya-1s/>

20) Конструктор запросов 1С [Электронный ресурс]: // Режим доступа:
http://v8.1c.ru/overview/Term_000000039.htm

21) Отчеты 1С [Электронный ресурс]: // Режим доступа:
<http://howknow1c.ru/rabota-v-1c/otchety-1s.html>

ПРИЛОЖЕНИЕ А

Программный код для кнопки «Рассчитать комплектующие»

Запрос = Новый Запрос;

```
Запросы = "ВЫБРАТЬ
|     Продукция.Номенклатура КАК Номенклатура,
|     Продукция.Количество КАК Количество
|ПОМЕСТИТЬ ВТПродукция
|ИЗ
|     &ТаблицаПродукции КАК Продукция
|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|     КомплектующиеНоменклатуры.Комплектующая КАК
Номенклатура,
|     СУММА(КомплектующиеНоменклатуры.Количество)
КАК Количество,
|     КомплектующиеНоменклатуры.ЕдиницаИзмерения,
|     СУММА(ВЫБОР
|             КОГДА
ЕСТЬNULL(ВТПродукция.Количество, 0) > 0
|             ТОГДА
ЕСТЬNULL(КомплектующиеНоменклатуры.Количество, 0) *
ВТПродукция.Количество
|             ИНАЧЕ 0
|     КОНЕЦ) КАК НеобходимоеКоличество,
|
```

КомплектующиеНоменклатуры.Комплектующая.ЕдиницаИзмерени
яМест.Коэффициент КАК КоличествоВУпаковке

|ПОМЕСТИТЬ ВТСводная

|ИЗ

| ВТПродукция КАК ВТПродукция

| ЛЕВОЕ СОЕДИНЕНИЕ

РегистрСведений.КомплектующиеНоменклатуры КАК

КомплектующиеНоменклатуры

| ПО ВТПродукция.Номенклатура =

КомплектующиеНоменклатуры.Номенклатура

|

|СГРУППИРОВАТЬ ПО

| КомплектующиеНоменклатуры.Комплектующая,

| КомплектующиеНоменклатуры.ЕдиницаИзмерения,

|

КомплектующиеНоменклатуры.Комплектующая.ЕдиницаИзмерени
яМест.Коэффициент

|;

|

////////////////////////////////////

|ВЫБРАТЬ

| ВТСводная.Номенклатура,

| ВТСводная.Количество,

| ВТСводная.ЕдиницаИзмерения,

| ВТСводная.НеобходимоеКоличество,

| ВТСводная.КоличествоВУпаковке,

| ЕСТЬNULL(ОстаткиТоваров.КоличествоОстаток, 0)

КАК КоличествоОстаток,

| ВЫБОР

| КОГДА

ЕСТЬNULL(ОстаткиТоваров.КоличествоОстаток, 0) -
 ВТСводная.НеобходимоеКоличество < 0
 | ТОГДА
 ЕСТЬNULL(ОстаткиТоваров.КоличествоОстаток, 0) -
 ВТСводная.НеобходимоеКоличество
 | ИНАЧЕ 0
 | КОНЕЦ * -1 КАК ТребуетсяЗаказать,
 | ВЫБОР
 | КОГДА
 ЕСТЬNULL(ОстаткиТоваров.КоличествоОстаток, 0) -
 ВТСводная.НеобходимоеКоличество < 0
 | ТОГДА
 ЕСТЬNULL(ОстаткиТоваров.КоличествоОстаток, 0) -
 ВТСводная.НеобходимоеКоличество
 | ИНАЧЕ 0
 | КОНЕЦ * -1 * ВТСводная.ЕдиницаИзмерения.Вес КАК
 Вес
 | ИЗ
 | ВТСводная КАК ВТСводная
 | ЛЕВОЕ СОЕДИНЕНИЕ (ВЫБРАТЬ
 | СвободныеОстаткиОстатки.Номенклатура
 КАК Номенклатура,
 |
 | СвободныеОстаткиОстатки.КоличествоОстаток КАК
 КоличествоОстаток
 | ИЗ
 |
 | РегистрНакопления.СвободныеОстатки.Остатки(, Склад В
 ИЕРАРХИИ (&Склад)) КАК СвободныеОстаткиОстатки) КАК
 ОстаткиТоваров

| ПО ВТСводная.Номенклатура =
ОстаткиТоваров.Номенклатура";

Запрос.УстановитьПараметр("Склад", СписокСкладов);
ТаблицаПродукции=Продукция.Выгрузить();
ТаблицаПродукции.Свернуть("Номенклатура", "Количество");
Запрос.УстановитьПараметр("ТаблицаПродукции",
ТаблицаПродукции);

Запрос.Текст = Запросы;
P3 = Запрос.Выполнить().Выгрузить();
Комплектующие.Загрузить(P3);

Для каждого СтрокаКомплектующих из Комплектующие цикл

ТребуемоеКоличествоВУпаковках=СтрокаКомплектующих.Требуе
тсяЗаказать/?(СтрокаКомплектующих.Номенклатура.ЕдиницаИзмерения
Мест.Коэффициент=0,1,СтрокаКомплектующих.Номенклатура.ЕдиницаИ
змеренияМест.Коэффициент);

ТребуемоеКоличествоВУпаковкахЦелое=Цел(СтрокаКомплектующ
их.ТребуетсяЗаказать/?(СтрокаКомплектующих.Номенклатура.ЕдиницаИ
змеренияМест.Коэффициент=0,1,СтрокаКомплектующих.Номенклатура.
ЕдиницаИзмеренияМест.Коэффициент));

ТребуемоеКоличествоВУпаковках=? (ТребуемоеКоличествоВУпако
вках-
ТребуемоеКоличествоВУпаковкахЦелое>0, ТребуемоеКоличествоВУпако
вкахЦелое+1, ТребуемоеКоличествоВУпаковкахЦелое);

СтрокаКомплекующих.ТребуетсяЗаказать=ТребуемоеКоличествоВ
Упаковках*?(СтрокаКомплекующих.Номенклатура.ЕдиницаИзмерения
Мест.Коэффициент=0,1,СтрокаКомплекующих.Номенклатура.ЕдиницаИ
змеренияМест.Коэффициент);

СтрокаКомплекующих.Вес=СтрокаКомплекующих.ТребуетсяЗа
казать*СтрокаКомплекующих.ЕдиницаИзмерения.Вес;

КонецЦикла;

КонецПроцедуры

ПРИЛОЖЕНИЕ Б

Программный код для кнопки «Сделать заказ»

ДокументЗаказ=Документы.ЗаказПоставщику.СоздатьДокумент();
 ЗаполнениеДокументов.ЗаполнитьШапкуДокумента(ДокументЗаказ
 , "Покупка");
 Для каждого СтрокаТоваров из Комплектующие цикл
 Если СтрокаТоваров.ТребуетсяЗаказать=0 тогда Продолжить;
 КонецЕсли;
 СтрокаТабличнойЧасти=ДокументЗаказ.Товары.Добавить();

 СтрокаТабличнойЧасти.Номенклатура=СтрокаТоваров.Номенклату
 ра;

 СтрокаТабличнойЧасти.Количество=СтрокаТоваров.ТребуетсяЗака
 зать;

 ОбработкаТабличныхЧастей.ПриИзмененииНоменклатурыТабЧаст
 и(СтрокаТабличнойЧасти, ДокументЗаказ);

 ОбработкаТабличныхЧастей.ЗаполнитьСтавкуНДСТабЧасти(Строк
 аТабличнойЧасти, ДокументЗаказ, "Приобретение");
 СтруктураШапкиДокумента = Новый Структура("Контрагент,
 ТипЦен, ДоговорКонтрагента, ДатаДокумента, ВалютаДокумента,
 УчитыватьНДС, СуммаВключаетНДС",
 ДокументЗаказ.Контрагент,
 ДокументЗаказ.ТипЦен, ДокументЗаказ.ДоговорКонтрагента,
 ДокументЗаказ.Дата,ДокументЗаказ.ВалютаДокумента,
 ДокументЗаказ.УчитыватьНДС,ДокументЗаказ.СуммаВключаетНДС);

ОбработкаТабличныхЧастей.ЗаполнитьЕдиницуЦенуПокупкиТабЧасти(СтрокаТабличнойЧасти, ДокументЗаказ, СтруктураШапкиДокумента, ДокументЗаказ.мВалютаРегламентированногоУчета, Ложь);

// Рассчитываем реквизиты табличной части.

ОбработкаТабличныхЧастей.РассчитатьСуммуТабЧасти(СтрокаТабличнойЧасти, ДокументЗаказ);

ОбработкаТабличныхЧастей.РассчитатьСуммуНДСТабЧасти(СтрокаТабличнойЧасти, ДокументЗаказ);

// Рассчитываем плановую себестоимость.

ОбработкаТабличныхЧастей.РассчитатьПлановуюСебестоимостьВСтрокеТабЧасти(СтрокаТабличнойЧасти, ДокументЗаказ, ДокументЗаказ.мВалютаРегламентированногоУчета);

// Заполним спецификацию по умолчанию.

СтрокаТабличнойЧасти.Спецификация =
УправлениеПроизводством.ОпределитьСпецификациюПоУмолчанию(СтрокаТабличнойЧасти.Номенклатура,
СтрокаТабличнойЧасти.ХарактеристикаНоменклатуры,
ДокументЗаказ.Дата, ДокументЗаказ.Подразделение);

КонецЦикла;

ДокументЗаказ.ПолучитьФорму().Открыть();

КонецПроцедуры

ПРИЛОЖЕНИЕ В

Плакаты презентации

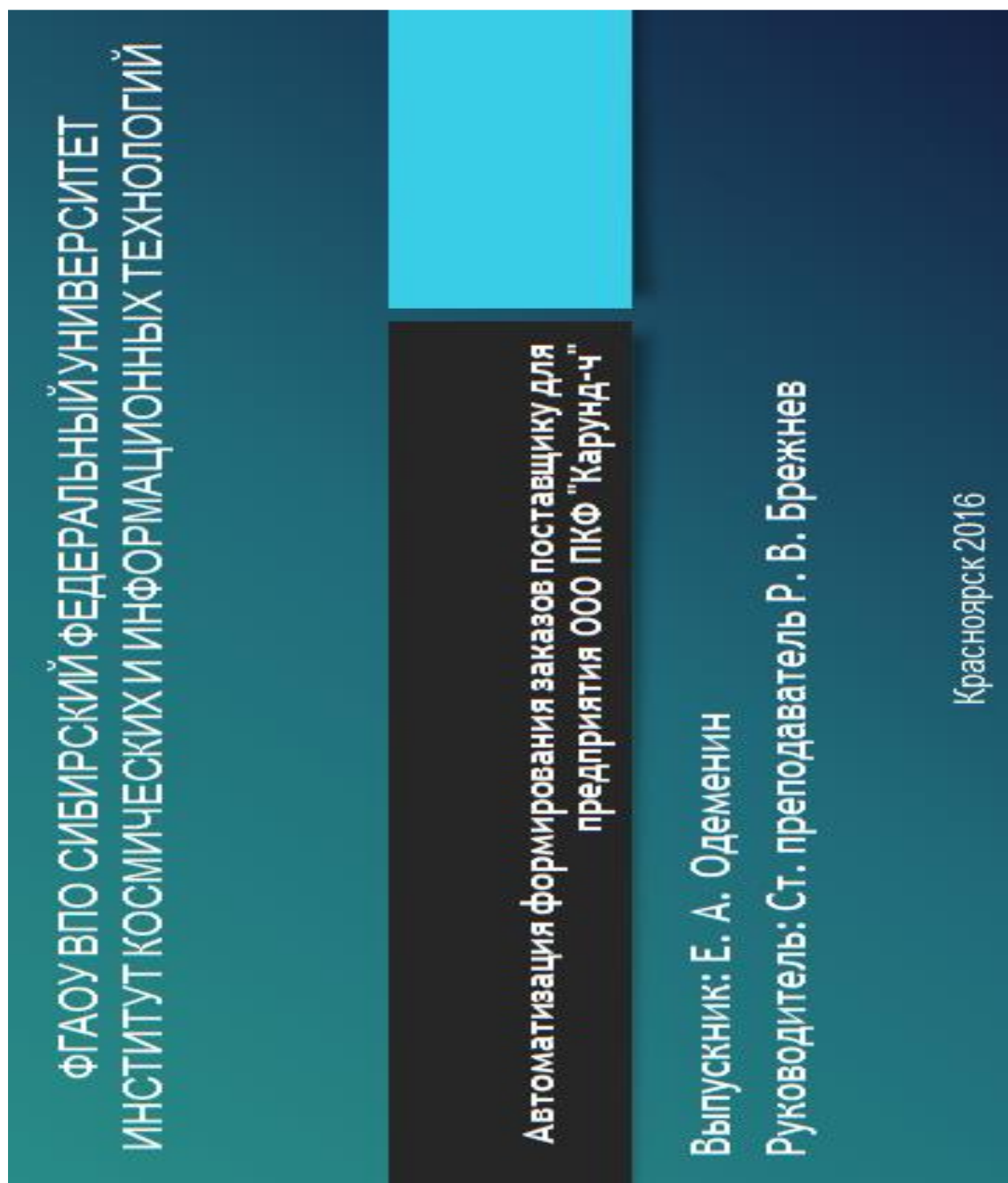


Рисунок В.1 — Плакат презентации № 1

Цель и задачи

2

Проектирование и реализация автоматизированной системы формирования заказов поставщику для предприятия ООО ПКФ «Карунд-Ч»

- Обзор существующих решений;
- Проектирование системы;
- Реализация системы.

Рисунок В.2 — Плакат презентации № 2

Функциональные возможности

3

- Разделение изделий на комплектующие;
- Подсчет необходимых для заказа изделий, исходя из остатков на складе;
- Выбор поставщика;
- Расчет цены;
- Отчет.

Рисунок В.3 — Плакат презентации № 3

Анализ существующих решений

4

Критерии	«1С: Предприятие»	«Галактика»	«Парус»	«БАР»
Доработка системы	5	1	1	3
Интерфейс	4	2	3	3
Модальность	4	5	5	4
Масштабируемость	5	4	5	5
Стоимость	5	3	3	1
ИТОГО:	23	15	17	16

Рисунок В.4 — Плакат презентации № 4

Диаграмма вариантов использования

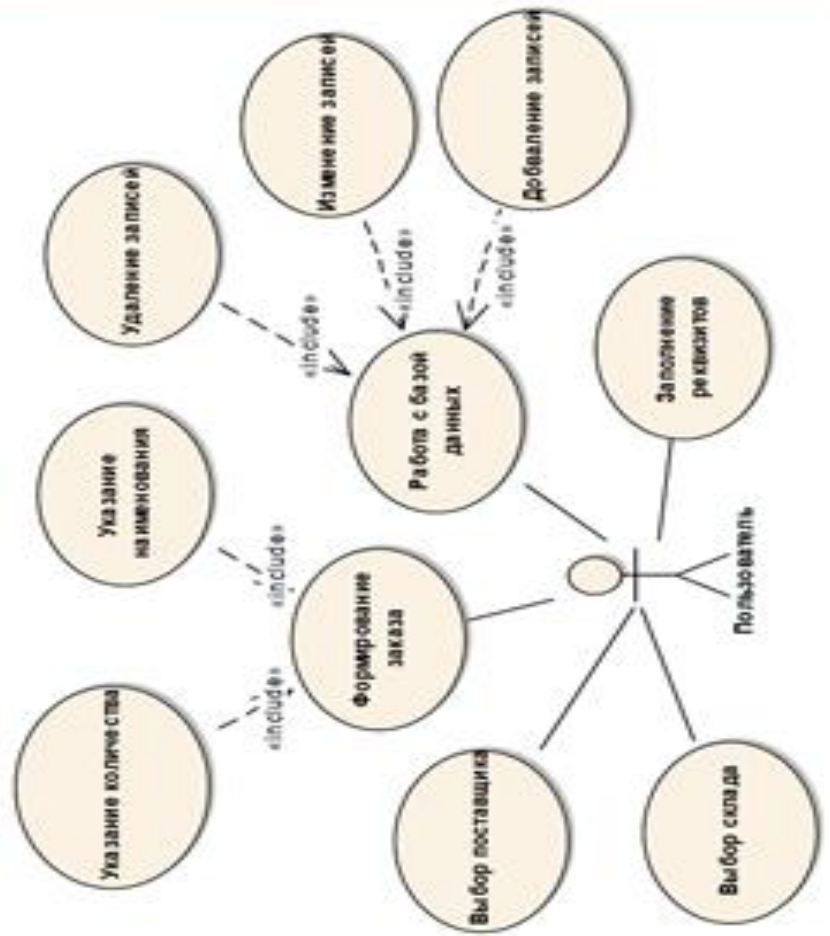


Рисунок В.5 — Плакат презентации № 5

Диаграмма вариантов использования

6

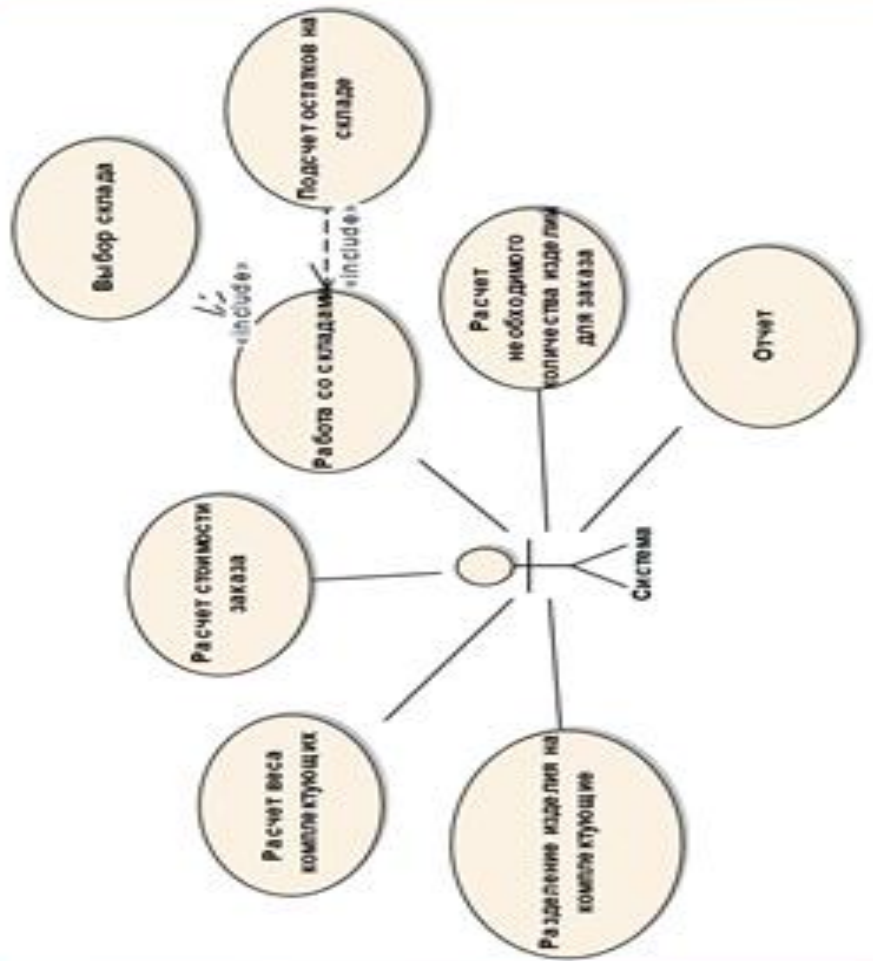


Рисунок В.6 — Плакат презентации № 6

Диаграмма деятельности

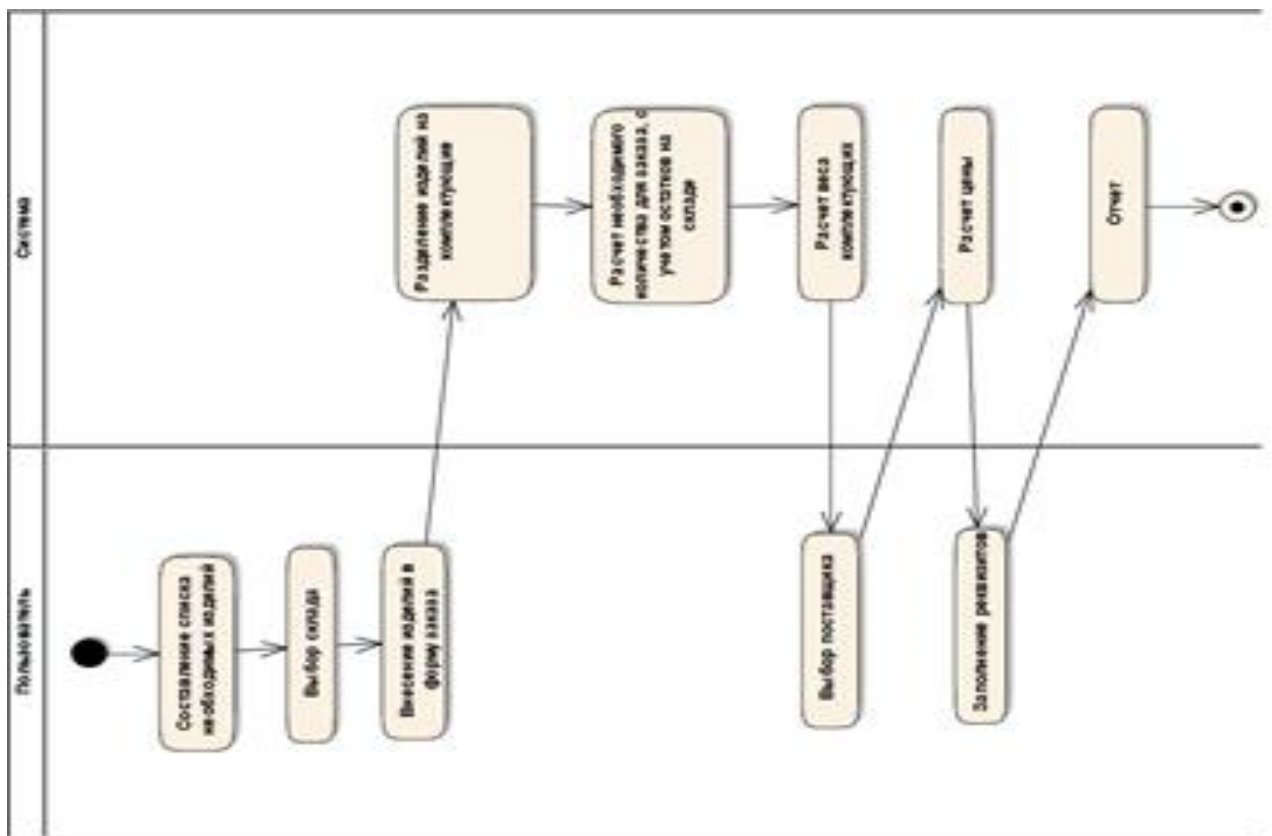


Рисунок В.7 — Плакат презентации № 7

Диаграмма классов

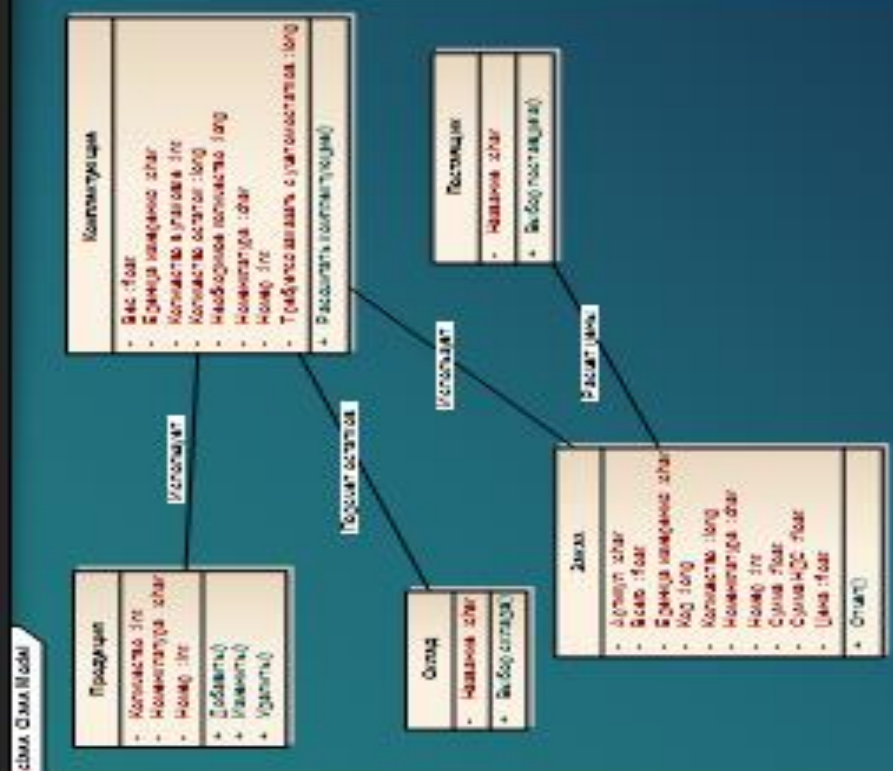


Рисунок В.8 — Плакат презентации № 8

Вкладка «Сделать заказ»

10

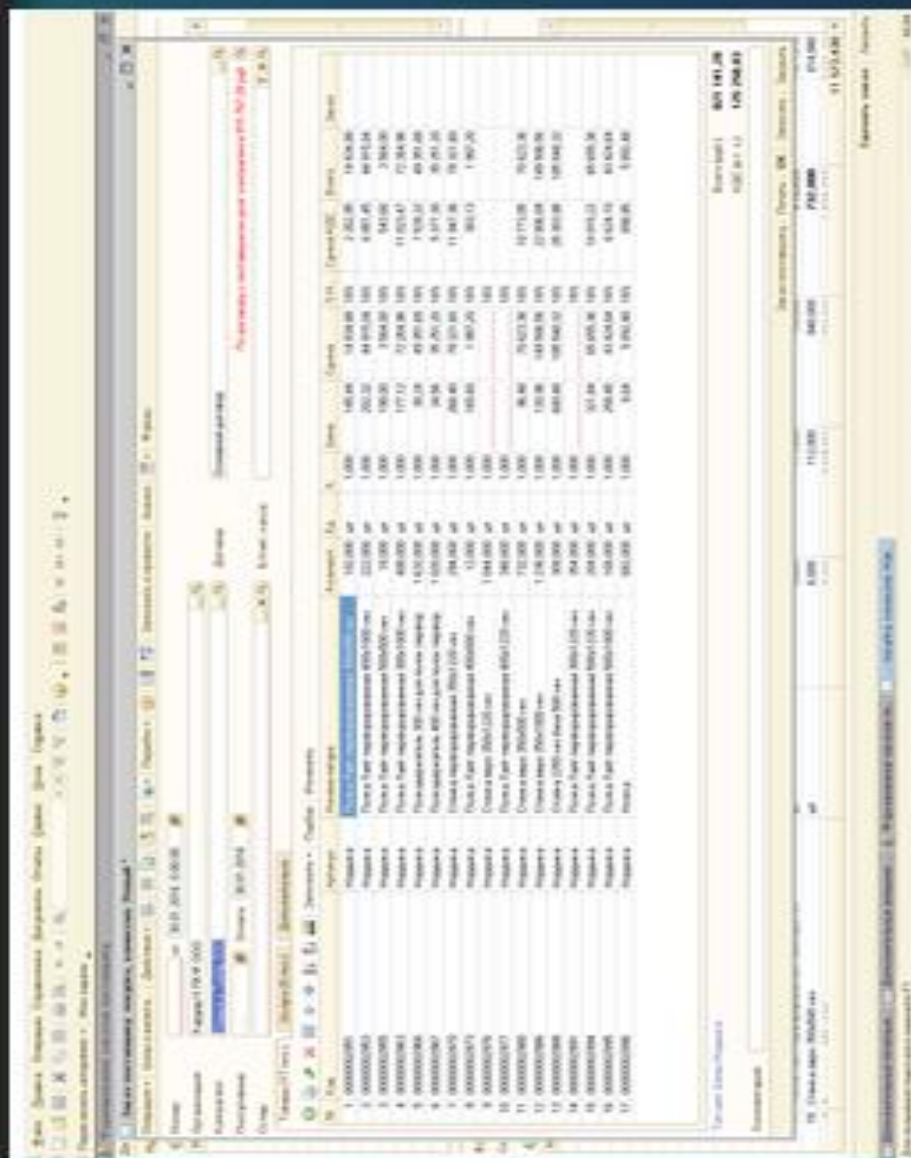


Рисунок В.10 — Плакат презентации № 10

Контрагенты

11

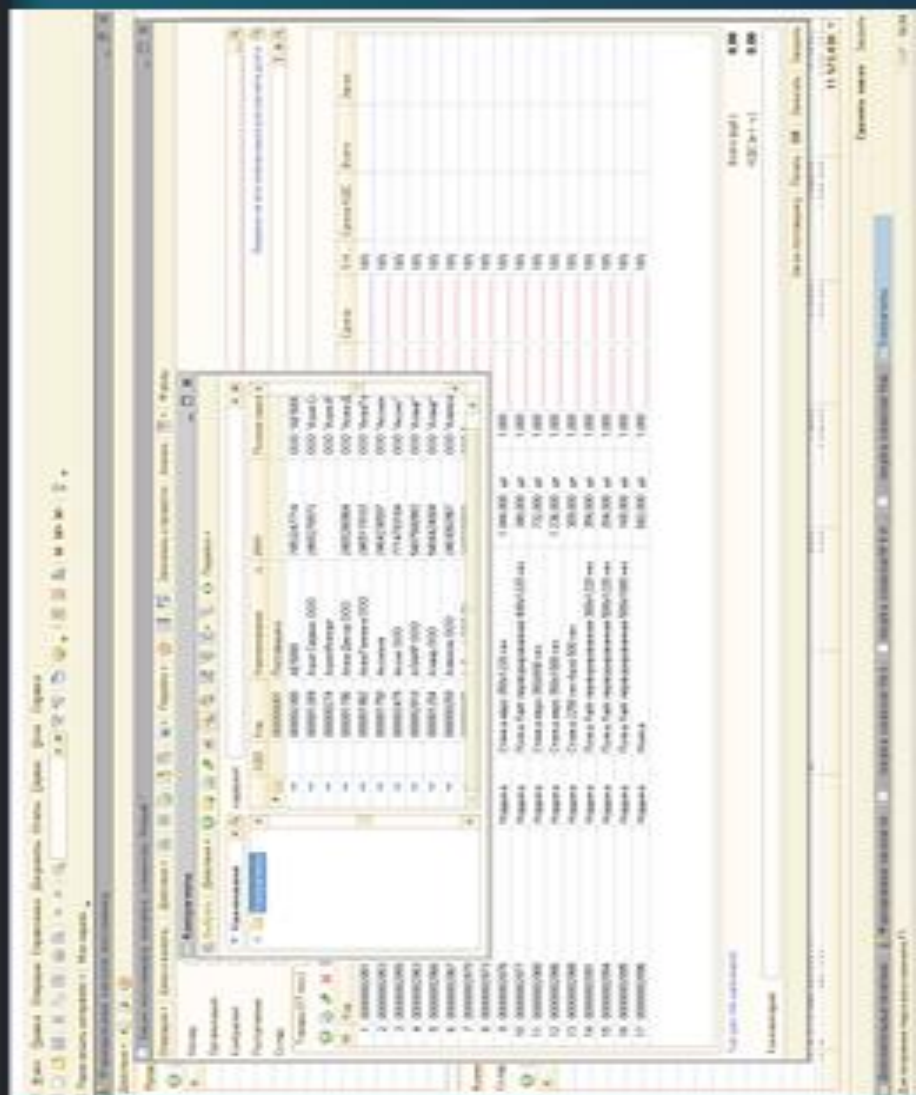


Рисунок В.11 — Плакат презентации № 11

Заказ поставки № 4 от 30 января 2016

Поставщик: ООО "Торговая Группа", ИНН 2222000000, КПП 222201001, 656041, Аугайский край, Барнаул г., Володарская ул., дом № 175 А, тел.: (3853) 24.91.63, 209.965, 209.962

Получатель: Общество с ограниченной ответственностью "Производственно-коммерческая фирма "Харьков Ч", ИНН 2444270001, КПП 244201001, 640813, Краснодарский край, Краснодар г., Ботаника Хмельницкого ул., дом № 4, корпус 11, тел.: 264.23.57

№	Артикул	Товары (факты, услуги)	Кол-во	Ед.	Цена	Сумма
1	Резерв	Пена для уборки/чистки 200х300 мм	152 шт		148.44	14 568.28
2	Резерв	Пена для уборки/чистки 400х100 мм	222 шт		207.21	46 011.62
3	Резерв	Пена для уборки/чистки 500х500 мм	15 шт		188.00	2 820.00
4	Резерв	Пена для уборки/чистки 100х100 мм	400 шт		177.12	70 848.00
5	Резерв	Антиржавчина 300 мм для пола, металл	1 131 шт		39.24	44 381.24
6	Резерв	Антиржавчина 400 мм для пола, металл	1 070 шт		34.86	37 288.20
7	Резерв	Сетка для уборки/чистки 300х120 мм	204 шт		268.40	54 765.60
8	Резерв	Пена для уборки/чистки 400х500 мм	31 шт		161.00	4 991.00
9	Резерв	Сетка для 300х120 мм	1 044 шт		36.45	38 038.20
10	Резерв	Пена для уборки/чистки 400х120 мм	301 шт		86.40	26 016.00
11	Резерв	Сетка для 300х500 мм	171 шт		170.00	29 070.00
12	Резерв	Сетка для 300х1000 мм	1 238 шт		120.00	148 560.00
13	Резерв	Сетка 2700 мм для пола 600 мм	309 шт		600.40	185 523.60
14	Резерв	Пена для уборки/чистки 300х120 мм	264 шт		227.40	59 993.60
15	Резерв	Пена для уборки/чистки 500х1200 мм	204 шт		250.40	51 081.60
16	Резерв	Пена для уборки/чистки 500х1000 мм	100 шт		250.40	25 040.00
17	Резерв	Итого	682 шт		3.80	2 607.60

Итого: 821 511,28
В том числе НДС: 125 250,80

Без НДС: 696 260,48

Всего от поставщика сумма заказа 676 000 рубль 28 копеек

Руководитель: _____ бухгалтер: _____

Рисунок В.12 — Плакат презентации № 12

Заключение

13

- Поставленные цели и задачи были выполнены, данная система успешно реализована и применяется на предприятии ООО ПКФ «Карунд-Ч».

Рисунок В.13 — Плакат презентации № 13