

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК
КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ И ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ

**ИНФОРМАЦИОННОЕ СОПРОВОЖДЕНИЕ ДЕЯТЕЛЬНОСТИ
ОРГАНИЗАЦИИ ПО РОЗНИЧНОЙ ПРОДАЖЕ ТОВАРОВ (НА
ПРИМЕРЕ МАГАЗИНА «TERRANOVA»)**

Выпускная квалификационная работа
обучающегося по направлению подготовки 09.04.03 Прикладная информатика
очной формы обучения, группы 07001633
Кизиловой Ольги Геннадиевны

Научный руководитель
профессор Петровский А.Б.

Рецензент
директор ООО «Шолди»
Шаломицкий С.В.

БЕЛГОРОД 2018

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 Теоретическая часть	5
1.1 Исследование литературных источников	5
1.2 Сравнительный анализ систем для управления бизнес-предприятием.....	12
1.3 Анализ деятельности предприятия.....	20
2 Исследовательская часть.....	26
3 Практическая часть.....	38
3.1 Программная реализация.....	38
3.2 Эффективность	51
ЗАКЛЮЧЕНИЕ	55
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	56
ПРИЛОЖЕНИЕ	62

ВВЕДЕНИЕ

Программный продукт «1С:Предприятие» каждый год все больше укрепляет свои позиции на современном российском рынке программ по управлению предприятием. По данным исследовательского центра TAdviser, доля данной отечественной программы от общего количества внедрённых решений составляет 31%.

Фирма «1С» предоставляет ряд продуктов использования в различных сферах деятельности организаций. Однако, клиентам часто требуются индивидуальные доработки для наиболее удобной и полной автоматизации бизнес-процессов. Продукт «1С:Предприятие» позволяет модифицировать механизмы работы и создавать новые, благодаря чему данный продукт становится возможным применять в любой сфере деятельности организации.

Актуальность обуславливается широкой востребованностью продуктов «1С:Предприятие» и индивидуальными особенностями работы многих фирм, которые требуется учитывать при доработке программного продукта или синхронизации данных с другими программами.

Объектом данной диссертационной работы является деятельность магазина «Terranova».

Предметом исследования является бизнес-процесс розничных продаж магазина «Terranova».

Целью является уменьшение времени простоев и коллизий при интеграции программ «1С:Предприятие» и CRM компании «Juce Development» в работе сотрудников отдела продаж на основе преобразования существующего механизма осуществления розничных продаж в «Terranova».

В диссертационной работе были поставлены следующие задачи:

- 1) изучить методы и подходы для управления бизнес-предприятием;
- 2) проанализировать существующие системы управления бизнес-предприятием, имеющие возможность интеграции со сторонними CRM-системами;

- 3) определить основной функционал и возможности синхронизации данных;
- 4) проанализировать работу организации, на основании чего составить модели «как есть» с помощью инструментальных средств представления информации;
- 5) создать и описать новый алгоритм выполнения бизнес-процесса розничной продажи;
- 6) разработать программу и протестировать ее работу;
- 7) рассчитать эффективность предложенного решения.

Научная новизна диссертационной работы состоит в следующем:

- 1) алгоритм осуществления интеграции «1С:Предприятие» и CRM компании «Juice Development»;
- 2) усовершенствованный механизма работы программы «1С:Предприятие», в соответствии с требованиями конкретного клиента (магазина «Terranova»).

Результатом диссертационной работы является доработанная программа. Практическая значимость заключается в следующем:

- 1) организация более качественного и удобного анализа продаж с использованием CRM;
- 2) построение работы отдела продаж в соответствии с требованиями руководства «Terranova»;
- 3) устранение коллизий и зависаний при оформлении реализации товаров.

Положения, выносимые на защиту:

- 1) результаты сравнительного анализа ERP-систем;
- 2) алгоритм осуществления продаж в магазине «Terranova»;
- 3) усовершенствованный механизм интеграции CRM и компании «Juice Development» и «1С:Предприятие».

Диссертационная работа содержит 61 страницу, 50 рисунков, 3 таблицы и 1 приложение.

1 Теоретическая часть

1.1 Исследование литературных источников

Основным механизмом для ведения качественного и доходного бизнес дела является программа, позволяющая отражать процессы организаций и выводить данные в виде удобном для произведения анализа их деятельности.

Первые системы для управления предприятием появились в 60-х годах 20 века. Это были MRP (Material Requirements Planning) [13] – система, автоматизирующая процесс планирования потребностей в сырье и материалах, предназначенных для производства. Данные системы позволили минимизировать издержки, полученные в результате складирования ресурсов предприятия. Предназначались они исключительно для производственной сферы.

В конце 70-х годов 20 века по причине недостатка функциональных возможностей в связи с увеличением потребностей предприятий на основе ранее описанной методологии MPS и FRP совместно с концепцией CRP были модернизированы программы MRP, которые получили название MRP II.

FRP (Finance Requirements Planning) [13] – способ, предназначенный для управления финансовыми ресурсами.

CRP (Capacity Requirements Planning) [13] – методология, позволяющая планировать производственные ресурсы.

Системы данного класса позволяют планировать и управлять сырьем, материалом, оборудованием и трудозатратами производства. Для удобства использования программ на основе данных технологий, основной интерфейс был разделен на несколько модулей, основываясь на функциональной принадлежности процесса. В результате были сформированы 7 разделов.

- 1) Бизнес-планирование, состоящие из процессов создания планов и получения производственного решения для предприятия.

- 2) Планирование продаж и деятельности.

3) Планирование производства, построенное на объемном или объемно-календарном плане. Данный модуль зачастую объединяют со следующим.

4) Формирование графика выпуска продукции. В данный раздел входят процессы, определяющие количество продукции и сроки их изготовления.

5) Планирование потребностей в материальных ресурсах.

6) Планирование производственных мощностей. Входящие в данный модуль процессы предназначены для сравнения имеющихся и требующихся ресурсов предприятия.

7) Оперативное управление производством.

В 90-х годах 20 века на общественное обозрение были выставлены ERP-системы, использующие в своем ядре технологию CSRM.

CSRP (Customer Synchronized Resources Planning) [13] – способ управления, который направлен на взаимодействие с клиентами. Работает с такими процессами, как поэтапное оформление заказов, создание планов, проектов и заданий, техническая поддержка.

ERP (Enterprise Resources Planning) [13] – метод управления корпоративными ресурсами.

Программы, использующие данные концепции, предназначены для организации работы с клиентами. ERP-системы расширили область применения автоматизации процессов предприятий, позволив управлять деятельностью не только производственных, но и бизнес-предприятий. Это сделало возможным использование продуктов данного типа так же в торговой, финансовой и сфере услуг.

Основной причиной создания данных систем было увеличение мощности вычислительных процессов и функциональных возможностей, т.к. предприятиям требовалось большое количество взаимосвязанных операций.

Для ERP-систем характерен определенный набор функций, которые можно назвать типовыми.

1) Ведение конструкторских и технологических спецификаций. Функции данного типа устанавливают состав товара. Так же включает в себя материальные ресурсы и операции, которые необходимы для производства изделия.

2) Управление спросом и формирование планов продаж и производства.

3) Планирование потребностей в материалах.

4) Управление запасами и закупочной деятельностью.

5) Планирование производственных мощностей. Определяет возможности производства или удовлетворения потребительского спроса, в зависимости от закупочного плана.

6) Финансовые функции. В эту группу входят функции финансового учета, управленческого учета, а также оперативного управления финансами.

7) Функции управления проектами.

Во второй половине 90-х годов 20 века был большой прирост появления новых компаний. Вследствие чего, необходима была оперативная связь с клиентами и поставщиками. Крупным компаниям требовались контроль работы сотрудников по отношению к заказчикам. Все эти потребности были учтены в CRM-системе.

Система управления взаимоотношениями с клиентами (CRM-система, Customer Relationship Management) [13] – прикладное программное обеспечение для организаций, предназначенное для автоматизации стратегий взаимодействия с заказчиками, в частности для повышения уровня продаж, оптимизации маркетинга и улучшения обслуживания клиентов путём сохранения информации о клиентах и истории взаимоотношений с ними, установления и улучшения бизнес-процессов и последующего анализа результатов.

Типовыми функциями для таких систем являются:

1) работа с информацией о клиентах;

2) управление операциями с торговыми представителями;

- 3) отражение контактов с клиентами и заказчиками посредством телефона, интернета, личной встречи и т.д.;
- 4) осуществление прогноза по введенным данным;
- 5) определение способов привлечения потенциальных клиентов;
- 6) управление заказами;
- 7) документооборот;
- 8) анализ и управление продажами;
- 9) информация и состав продукта;
- 10) маркетинговые операции.

В 1999 году была произведена модернизация ERP-систем до модели второй версии. ERP II дополнены такими функциями, как взаимоотношения с клиентами, организацией последовательности поставок и возможностью ведения работы предприятия через интернет.

В конце 20-го века так же были выпущены следующие узкоспециализированные системы:

- 1) CAD (Computer-Aided Design) с функциями проектирования;
- 2) CAM (Computer-Aided Manufacturing), работающая технологии производства;
- 3) PDM (Project Data Management) для документооборота;
- 4) CAE (Computer-Aided Engineering) для инженерных расчетов;
- 5) MES (Management Execution System), предназначена для ведения учета на цехах или в производстве в целом;
- 6) SCADA (Supervisory Control And Data Acquisition System) служит для сбора данных и управления технологическими процессами.

Прежде чем приступить к подробному сравнению технологий по предоставляемым функциям, необходимо определить их сильные и слабые стороны, а также сложившуюся ситуацию на рынке.

Поскольку технология MRP предназначена для производственных предприятий, далее рассматриваться она не будет.

ERP и CRM-системы зачастую объединяют, внося соответствующий

модуль. Ниже будет произведено сравнение технологий, а не программ на их основе. В связи с чем системы рассматриваются как отдельные, независимые концепции.

Достоинства и недостатки анализируемых технологий представлены в таблице 1.1.

Таблица 1.1 – Характерные черты бизнес технологий

Параметр	CRM	ERP
1	2	3
Достоинства	Организация работы call-центра или другого отдела, требующего связи любого вида с клиентами	Отражение операций с поставщиками и заказчиками
	Широкий спектр функциональных возможностей	
	Создание стратегий управления бизнесом	
	Полный контроль работы сотрудников.	Разделение системы на логические модули, позволяющие работать с бухгалтерским учетом, управленческими и финансовыми процессами.
	Относительно небольшая стоимость продуктов данной серии, или же бесплатные версии	Контроль взаимосвязанных видов деятельности
	Интегрируемость с технологиями различного уровня	Интеграция с низкоуровневыми системами управления
	Удобное и настраиваемое под конкретного пользователя рабочее место	Наличие систем разных масштабов под размер предприятия
	Различные виды вывода данных	
	Централизованное хранение данных	
	Разграничение прав доступа к базе	
	Обмен данными с другими системами	
	Быстрота внедрения	Адаптируемость
	Недостатки	Невозможность адаптации к конкретным задачам
Сложность при формировании отчетов под конкретные критерии		Длительное время внедрения

Продолжение таблицы 1.1

1	2	3
Недостатки	Краткосрочные затраты на обучение персонала	Длительные дополнительные затраты на техническую поддержку и обучение персонала
	Затруднения при переносе данных	
	Избыточная информация	
	Трудности прогнозирования	Возможно ежегодное продление срока лицензии

Определив основные плюсы и минусы при внедрении данных технологий, необходимо перейти к сравнению систем по функциональности и выявлению наиболее оптимального варианта для ведения бизнеса. В таблице 1.2 представлен анализ систем по основным возможностям. Наличие или отсутствие параметра обозначается «-» и «+» соответственно. Критериями, по которым производится сравнение CRM и ERP-систем по функциональным возможностям, послужили наиболее важные операции, выполняющиеся для грамотного ведения бизнеса.

Таблица 1.2 – Ключевые параметры для ведения бизнеса

Параметр	Функция	CRM	ERP
1	2	3	4
Кадры	Управление персоналом	+	+
	Контроль работы сотрудников	+	-
	Создание заданий по определенному сотруднику с указанием сроков выполнения	+	-
	Рабочий график	+	+
	Оповещение о мероприятиях, важных событиях	+	-
	Анализ по кадрам	+	+
	Закрепление сотрудника за клиентом	+	-
Производство	Учет производственной продукции	-	+
	Отражение операций и затрат по производству	-	+
	Анализ по производству	-	+
Ресурсы	Учет товаров на складах	-	+
	Отражение операций с товарами	-	+
	Отражение основных средств и операций, связанных с ними	-	+
	Анализ по материально-техническому оснащению	-	+

Продолжение таблицы 1.2

1	2	3	4
Маркетинг	Отражение маркетинговых мероприятий	+	+
	Анализ по маркетингу	+	+
Управление	Удобное централизованное, мобильное хранение и поиск данных	+	+
	Отражение операций бухгалтерского и налогового учета	-	+
	Отражение финансовых операций	+	+
	Отражение управленческих операций	+	+
	Анализ по управлению	+	+
Торговля	Отражение и анализ операций с клиентами	+	+
	Ведение базы клиентов	+	+
	Ведение базы поставщиков	-	+
	Отражение и анализ операций с поставщиками	-	+
	Стандартизированные государственные отчеты	-	+
	Анализ по торговле	+	+
Финансы	Отражение финансовых операций по сотрудникам	-	+
	Отражение финансовых операций по производству	-	+
	Отражение финансовых операций по ресурсам	-	+
	Отражение финансовых операций по маркетингу	+	+
	Отражение финансовых операций по торговле	+	+
	Отслеживание задолженностей клиентов	+	+
	Отслеживание задолженностей поставщиков	-	+
	Отражение прочих затрат	+	+
	Анализ по финансам	+	+
Итого		20	32

Таким образом, для анализа технологий были взяты 36 основных действий, распределенных на 7 составляющих, отражаемых на предприятии. По результатам сравнения было выявлено, что CRM-системы уступают ERP-системам по функциональным возможностям и охвату бизнес операций. Это связано с тем, что основное предназначение CRM – управление самим бизнес-предприятием, его структурой и основными операциями, выполняемыми сотрудниками. ERP-системы же предназначены для непосредственного ведения бизнеса, что обуславливает наличие всех вышеописанных модулей.

1.2 Сравнительный анализ систем для управления бизнес-предприятием

Сравнение систем производится с использованием метода анализа иерархии (МАИ) [18] – математический инструмент системного подхода к сложным проблемам принятия решений.

Сравнение выполнялось с использованием Excel 2010 и на основе субъективных суждений эксперта.

Порядок применения МАИ:

1) Построение качественной модели проблемы в виде иерархии, включающей цель, альтернативные варианты достижения цели и критерии для оценки качества альтернатив.

2) Определение приоритетов всех элементов иерархии с использованием метода парных сравнений.

3) Синтез глобальных приоритетов альтернатив путём линейной свертки приоритетов элементов на иерархии.

4) Проверка суждений на согласованность.

5) Принятие решения на основе полученных результатов.

Анализ на основе выбранного метода выполнялся поэтапно.

Выполнение первого этапа – построение модели.

ERP-системы для сравнения были выбраны на основании данных веб-портала TAdviser, тщательно собирающего и анализирующего информацию на основе платформы Wiki. Данный сайт предоставляет данные, определяющие наиболее востребованные ERP-системы за 2017 г., а также лидеров по количеству ERP-проектов. Для наглядности представления полученные сведения были представлены в графическом виде на рисунках 1.1 и 1.2.

САМЫЕ ПОПУЛЯРНЫЕ ERP-СИСТЕМЫ*

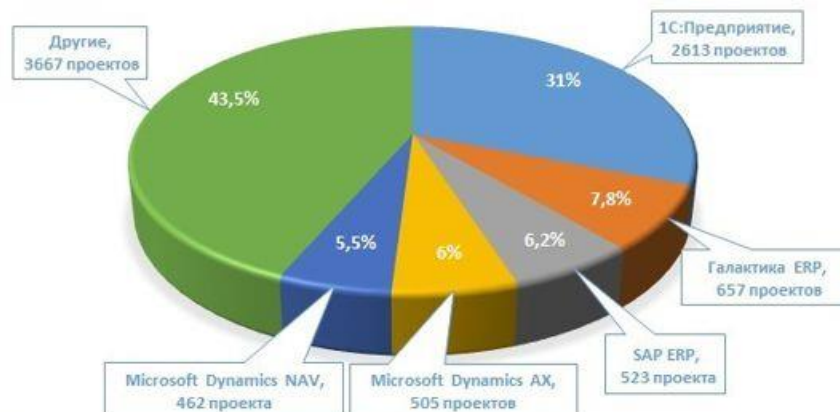


Рисунок 1.1 – Данные портала TAdviser по популярности систем

ВЕНДОРЫ-ЛИДЕРЫ ПО КОЛИЧЕСТВУ ERP-ПРОЕКТОВ*

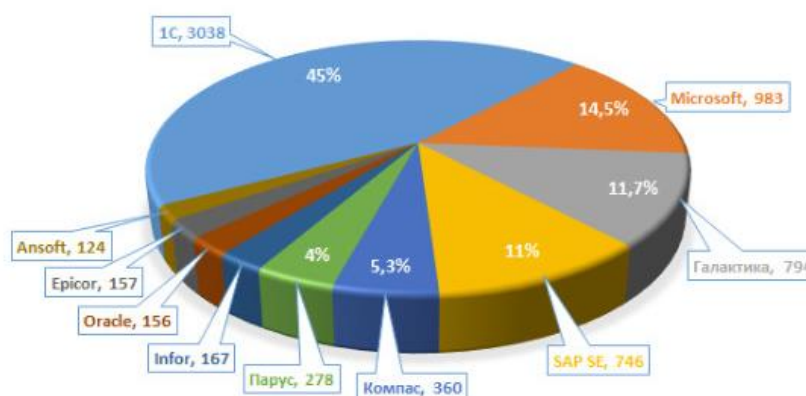


Рисунок 1.2 – Данные портала TAdviser по количеству внедрений

Анализ будет производиться на основе отобранных по востребованности среди полного объема потребителей, не зависимо от их сферы деятельности. Также при выборе программ для внедрения в интернет-магазин учитывалось наличие функциональных возможностей для отражения бизнес-процессов в следующих сферах деятельности фирмы:

- 1) управленческий учет;
- 2) финансовый учет;
- 3) кадровый учет.

Проанализировав полученные данные, функциональные возможности и сопоставив данные с целью исследования, был составлен список продуктов

компаний и продуктов, занимающих лидирующие позиции среди ERP-систем, позволяющих вести розничную торговлю. Все данные были сгруппированы по виду фирмы. Результат представлен в таблице 1.3.

Таблица 1.3 – ERP-системы

Программа	Версия
Отечественные	
1С:Предприятие	Розница 1
Галактика	Торговля
Парус	Модуль Комплекс
Зарубежные	
Oracle	Oracle Business Intelligence
Microsoft	Microsoft Dynamics AX
Sap	Sap r/3

Для сравнения ERP-продуктов были определены основные требования, опирающиеся на бизнес-процессы и вид работы предприятия. К таким критериям были отнесены следующие данные:

- 1) общая стоимость программы;
- 2) интеграция с системой CRM;
- 3) период внедрения;
- 4) период обучения сотрудников;
- 5) быстрое действие;
- 6) открытость программного кода;
- 7) наличие русского интерфейса.

Выполнение второго этапа – попарного сравнения критериев представлено на рисунке 1.3.

На данном этапе критериям выставляется сравнительная оценка относительно остальных критериев по шкале от 1 до 9, где оценки имеют следующее значение:

- 1) равная важность – 1;
- 2) умеренное превосходство – 3;
- 3) значительное превосходство – 5;

- 4) сильное превосходство – 7;
- 5) очень сильное превосходство – 9;
- б) в промежуточных случаях ставятся четные оценки: 2, 4, 6, 8 (например, 4 – между умеренным и значительным превосходством).

После сравнения были получены данные о:

- 1) средних геометрических – числа, которыми можно заменить каждое из этих чисел так, чтобы их произведение не изменилось (формула 1.1);
- 2) нормализованных векторах приоритетов – приведение чисел к виду, в котором их сумма будет равняться единице (формула 1.2);

Получив результаты вычисления производим проверку:

- 1) индекс согласованности (ИС) локальных приоритетов – степени нарушения численной (кардинально) и транзитивной (порядковой) согласованности (формулы 1.3-1.4);
- 2) отношение согласованности (ОС) элементов матрицы – степень корректности выставленных количественных оценок приоритетов (формула 1.5).

$$A_i = \sqrt[n]{\text{произведение элементов } 1 - \text{ой строки}} \quad (1.1)$$

$$\text{НВП}_i = \frac{a_i}{\sum a_i} \quad (1.2)$$

$$\lambda = \sum(\text{сумма элементов } i - \text{го столбца} * i - \text{ый компонент НВП}) \quad (1.3)$$

$$\text{ИС} = \frac{\lambda - n}{n - 1}, \quad (1.4)$$

где n – количество критериев.

$$\text{ОС} = \frac{\text{ИС}}{\text{ПСС}} \quad (1.5)$$

Оценки в матрице считаются согласованными, если $\text{ОС} \leq 10\text{-}15\%$, в

противном случае их надо пересматривать.

Критерии									
	Стоимость	Интеграция с CRM	Период внедрения	Период обучения	Быстродействие	Открытый код	Русский интерфейс	Геометр	НВП
Стоимость	1	3	5	5	3	3	2	2,80	0,33
Интеграция с CRM	1/3	1	3	3	2	2	2	1,57	0,18
Период внедрения	1/5	1/3	1	3	1/3	1/3	1/3	0,50	0,06
Период обучения	1/5	1/3	1/3	1	1/3	1/5	1/5	0,31	0,04
Быстродействие	1/3	1/2	3	3	1	1/2	2	1,06	0,12
Открытый код	1/3	1/2	3	5	2	1	1/2	1,14	0,13
Русский интерфейс	1/2	1/2	3	5	1/2	2	1	1,21	0,14
$\lambda(\max)$	7,50						Сумма	8,59	1
ИС	0,08								
ОС	0,06								

Рисунок 1.3 – Результат сравнения критериев

В результате сравнительного анализа параметров наиболее приоритетными являются «Стоимость» и «Интеграция с CRM».

Выполнение третьего этапа – попарное сравнение альтернатив представлено на рисунках 1.4-1.10.

Принцип выполнения данного этапа такой же, как и в предыдущем этапе. Сравнение производится среди альтернатив.

Стоимость								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	2	3	6	5	7	3,29	0,37
Галактика	1/2	1	3	6	5	7	2,61	0,29
Парус	1/3	1/3	1	5	4	7	1,58	0,18
Oracle	1/6	1/6	1/5	1	1/2	5	0,49	0,06
Microsoft	1/5	1/5	1/4	2	1	5	0,68	0,08
Sap	1/7	1/7	1/7	1/5	1/5	1	0,22	0,02
$\lambda(\max)$	6,47						Суммы	8,87
ИС	0,09							
ОС	0,08							

Рисунок 1.4 – Результат сравнения альтернатив по критерию «Стоимость»

В результате сравнительного анализа вариантов по параметру «Стоимость» наиболее подходящим является «1С:Предприятие».

Интеграция с CRM								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	7	3	5	7	5	3,93	0,43
Галактика	1/7	1	1/7	1/7	1	1/7	0,27	0,03
Парус	1/3	7	1	1/5	7	1/5	0,93	0,10
Oracle	1/5	7	5	1	7	1	1,91	0,21
Microsoft	1/7	1	1/7	1/7	1	1/7	0,27	0,03
Sap	1/5	7	5	1	7	1	1,91	0,21
$\lambda(\max)$	7,178592					Суммы	9,23	1,00
ИС	0,235718							
ОС	0,190096							

Рисунок 1.5 – Результат сравнения альтернатив по критерию «Интеграция с CRM»

В результате сравнительного анализа вариантов по параметру «Интеграция с CRM» наиболее подходящим является «1С:Предприятие».

Период внедрения								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	3	3	7	5	7	3,61	0,42
Галактика	1/3	1	1	5	3	5	1,71	0,20
Парус	1/3	1	1	5	3	5	1,71	0,20
Oracle	1/7	1/5	1/5	1	1/3	1	0,35	0,04
Microsoft	1/5	1/3	1/3	3	1	3	0,76	0,09
Sap	1/7	1/5	1/5	1	1/3	1	0,35	0,04
$\lambda(\max)$	6,18					Суммы	8,50	1,00
ИС	0,04							
ОС	0,03							

Рисунок 1.6 – Результат сравнения альтернатив по критерию «Период внедрения»

В результате сравнительного анализа вариантов по параметру «Период внедрения» наиболее подходящим является «1С:Предприятие».

Период обучения								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	3	2	7	5	5	3,19	0,37
Галактика	1/3	1	1/3	7	5	5	1,64	0,19
Парус	1/2	3	1	7	5	5	2,53	0,29
Oracle	1/7	1/7	1/7	1	1/3	1/3	0,26	0,03
Microsoft	1/5	1/5	1/5	3	1	3	0,64	0,07
Sap	1/5	1/5	1/5	3	1/3	1	0,45	0,05
$\lambda(\max)$	6,48					Суммы	8,71	1,00
ИС	0,10							
ОС	0,08							

Рисунок 1.7 – Результат сравнения альтернатив по критерию «Период обучения персонала»

В результате сравнительного анализа вариантов по параметру «Период обучения персонала» наиболее подходящим является «1С:Предприятие».

Быстродействие								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	3	1/3	1/3	1/5	1/5	0,49	0,06
Галактика	1/3	1	1/3	1/5	1/7	1/7	0,28	0,03
Парус	3	3	1	1/3	1/5	1/5	0,70	0,08
Oracle	3	5	3	1	1/3	1/3	1,31	0,15
Microsoft	5	7	5	3	1	1	2,84	0,34
Sap	5	7	5	3	1	1	2,84	0,34
$\lambda(\max)$	6,22					Суммы	8,45	1,00
ИС	0,04							
ОС	0,04							

Рисунок 1.8 – Результат сравнения альтернатив по критерию «Быстродействие»

В результате сравнительного анализа вариантов по параметру «Быстродействие» наиболее подходящими являются «Microsoft» и «Sap».

Открытый код								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	7	3	3	7	3	3,31	0,38
Галактика	1/7	1	1/7	1/7	1	1/7	0,27	0,03
Парус	1/3	7	1	1	7	1	1,59	0,18
Oracle	1/3	7	1	1	7	1	1,59	0,18
Microsoft	1/7	1	1/7	1/7	1	1/7	0,27	0,03
Sap	1/3	7	1	1	7	1	1,59	0,18
$\lambda(\max)$	6,25					Суммы	8,64	1,00
ИС	0,05							
ОС	0,04							

Рисунок 1.9 – Результат сравнения альтернатив по критерию «Открытость программного кода»

В результате сравнительного анализа вариантов по параметру «Открытость программного кода» наиболее подходящим является «1С:Предприятие».

Русский интерфейс								
	1С	Галактика	Парус	Oracle	Microsoft	Sap	Геометр	НВП
1С	1	1	1	7	3	3	1,99	0,26
Галактика	1	1	1	7	3	3	1,99	0,26
Парус	1	1	1	7	3	3	1,99	0,26
Oracle	1/7	1/7	1/7	1	1/5	1/5	0,22	0,03
Microsoft	1/3	1/3	1/3	5	1	1	0,75	0,10
Sap	1/3	1/3	1/3	5	1	1	0,75	0,10
$\lambda(\max)$	6,06					Суммы	7,72	1,00
ИС	0,01							
ОС	0,01							

Рисунок 1.10 – Результат сравнения альтернатив по критерию «Наличие русского интерфейса»

В результате сравнительного анализа вариантов по параметру «Наличие русского интерфейса» наиболее подходящими являются «1С:Предприятие», «Галактика» и «Парус».

Выполнение четвертого этапа – определение общего веса для каждой альтернативы.

Вес варианта – это величина, определяющее приоритет каждого из вариантов. Рассчитывается по формуле 1.6.

$$\text{Вес} = \sum \text{НВП}_i(\text{по критерию}) * \text{НВП}_i(\text{по альтернативе}) \quad (1.6)$$

Где НВП – нормализованный вектор приоритетов.

По завершению данного этапа определяется наиболее подходящий по выдвинутым критериям вариант.

Общий приоритет					
1С	Галактика	Парус	Oracle	Microsoft	Sap
0,3309787	0,164372	0,1698324	0,107051	0,09775	0,130016

Рисунок 1.11 – Результат сравнения общих приоритетов альтернатив

Исходя из данных, представленных на рисунке 1.11, наиболее предпочтительной системой для использования по критериям организации оказалась система «1С:Предприятие» в конфигурации «Розница» 1.0.

1.3 Анализ деятельности предприятия

Магазин «Terranova» города Белгород относится к сети товарных точек, принадлежащих ИП Свечкарь С.А..

ИП Свечкарь Станислав Алексеевич зарегистрирован 30 апреля 1998 г. регистратором инспекция федеральной налоговой службы по г. Белгороду. Основным видом деятельности является «Торговля розничная одеждой в специализированных магазинах».

Молодежная марка одежды «Terranova» родом из Италии. Сегодня магазины «Terranova» можно встретить в более чем 30-ти странах мира, в России в том числе. Одежда представлена в стиле wild casual stile – футболки, топы рубашки, джинсы, платья, вещи для спорта, сумки, обувь и аксессуары для

молодых людей и девушек. Основным принципом «Terranova» – это свежесть, новизна.

Для обеспечения большего спроса на товары, а также поддержки конкурентоспособности проводит акции, использует бонусные карты, делает скидки для своих клиентов и начисляет баллы для предстоящих покупок в данном магазине.

На сегодняшний день для организации продаж предприятие использует «1С:Предприятие» конфигурации «Розница» 1.0. Данные из базы отправляются в CRM «Sail Play», которая является веб-приложением. При продаже из «Sail Play» поступают отобранные личные данные по клиенту о возможных скидках, бонусах, доступном количестве баллов для списания и начисления.

При проведении документа «Чек ККМ» со сведениями о реализациях товаров определенному клиенту в программе «1С:Предприятие» данные по чеку регистрируются в CRM, что позволяет удобно и быстро производить анализ отдела продаж.

Для анализа принципа работы предприятия следует определить входящие, выходящие, управленческие потоки и механизмы. Моделирование потоков данных «Как есть» разрабатывалось посредством программы BPWin. При моделировании использовались методологии IDEF0 и IDEF3.

BPwin [27] – это незаменимый инструмент менеджеров и бизнес-аналитиков, в которую включена поддержка диаграмм потоков данных и методики IDEF3 (BPwin Professional), является мощным средством моделирования процессов при создании корпоративных информационных систем.

IDEF0 [29] – методология функционального моделирования (англ. function modeling) и графическая нотация, предназначенная для формализации и описания бизнес-процессов.

IDEF3 [29] – методология, описывающая процессы в определенной последовательности, а также объекты, участвующие совместно в одном процессе.

Реализация товаров происходит следующим образом. Клиент выбирает вещи из представленного ассортимента в торговом помещении. Когда покупатель определился со списком покупок, он обращается к продавцу для регистрации сделки с магазином и оплаты товаров. Продавец регистрирует продажу в «1С:Предприятие» и после получения денежных средств от клиента операция передается в «CRM Sail Play».

Вся работа производится в соответствии с инструкцией по работе с программой, уставом фирмы и законодательством Российской Федерации.

На рисунке 1.12 представлены основные потоки, участвующие в бизнес-процессе продаж, где:

- 1) входные потоки – «Информация по приобретаемым товарам клиента» и «Информация о клиенте»;
- 2) выходные потоки – «Информация по реализации товаров»;
- 3) механизмы – «1С:Предприятие», «CRM «Sail Play»» и «Продавец»;
- 4) управление – «Инструкции по работе с программой», «Устав фирмы» и «Законодательство Российской Федерации».

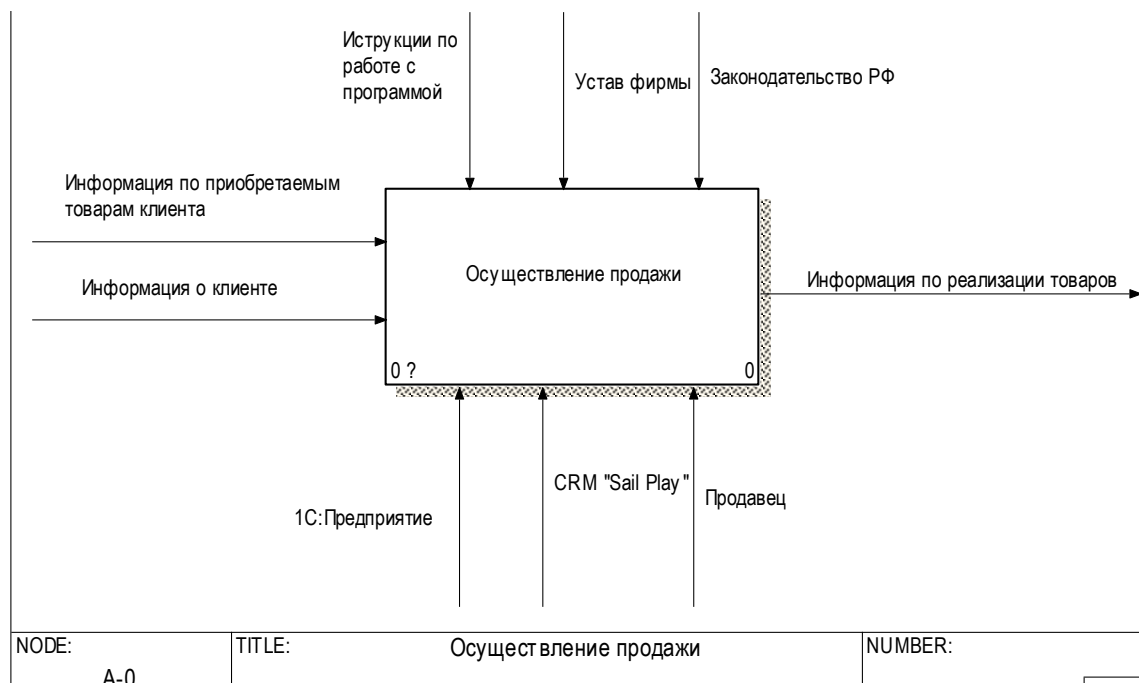


Рисунок 1.12 – Диаграмма «Осуществление продажи»

На рисунке 1.13 представлена декомпозиция бизнес-процесса «Осуществление продаж». Далее будут рассматриваться только 2 блока – «Использование модуля Sail Play» и «Проведение продажи», поскольку данные этапы будут реорганизованы.

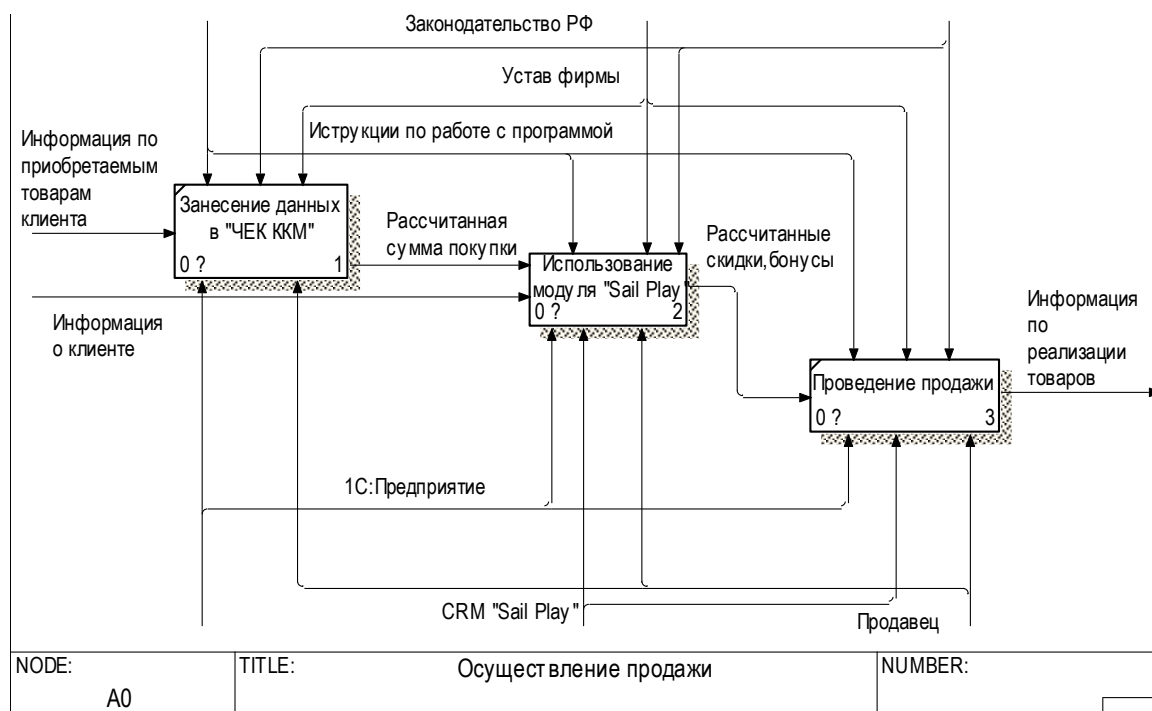


Рисунок 1.13 – Диаграмма декомпозиции «Осуществление продаж»

На рисунке 1.14 описано использование модуля Sail Play. На данной диаграмме наглядно представлены существующие недочеты механизма. Этап «Расчет доступных бонусов для списания» выполняется только программой «1С:Предприятие», без участия CRM. В «Рознице» программно указан процент равный 0,5 от которого ведется расчет. Данный процент в магазине Terranova не является статичным. Он должен быть динамическим и указываться в модуле CRM.

На этапе «Определение суммы скидки (списанные и начисленные бонусы)» продавец задает данные скидки. В программе «1С:Предприятие» не предусмотрен контроль вводимых данных.

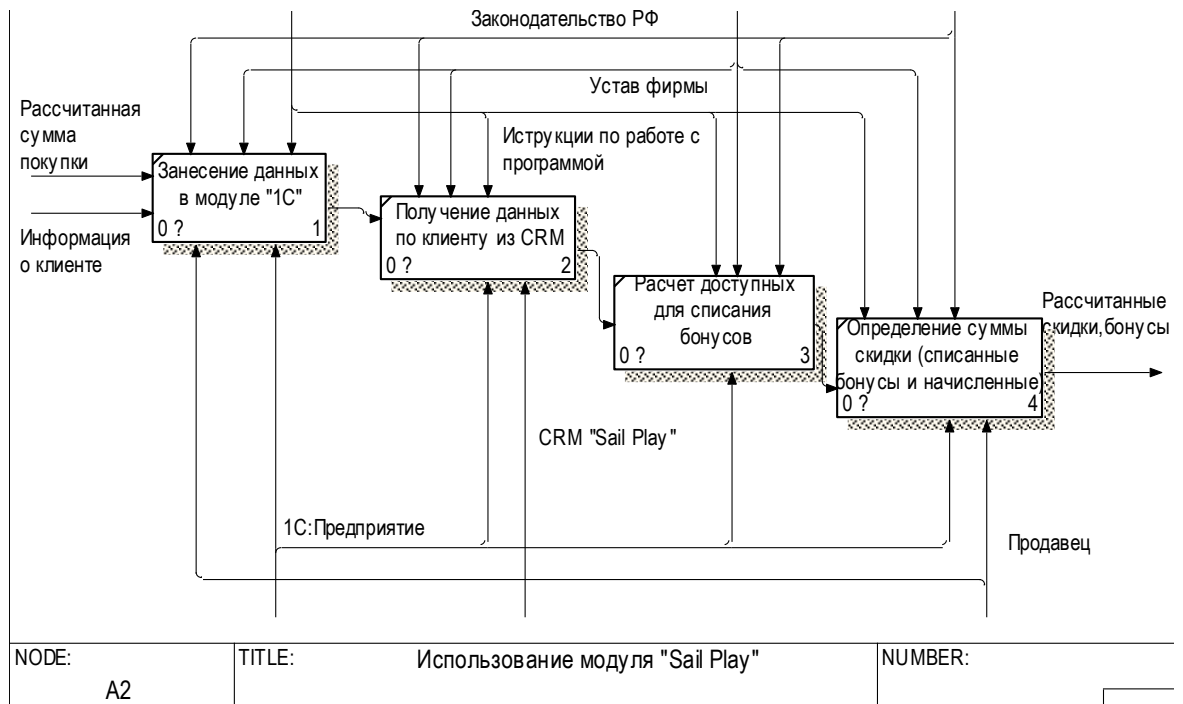


Рисунок 1.14 – Декомпозиция «Использование модуля Sail Play»

На рисунке 1.15 описан механизм непосредственного отражения реализации товаров в программах. Этап выполнения отложенных транзакций выполняется пользователем по нажатию на соответствующую кнопку. Исходя из вышесказанного получаем, что данный процесс не оптимизирован.

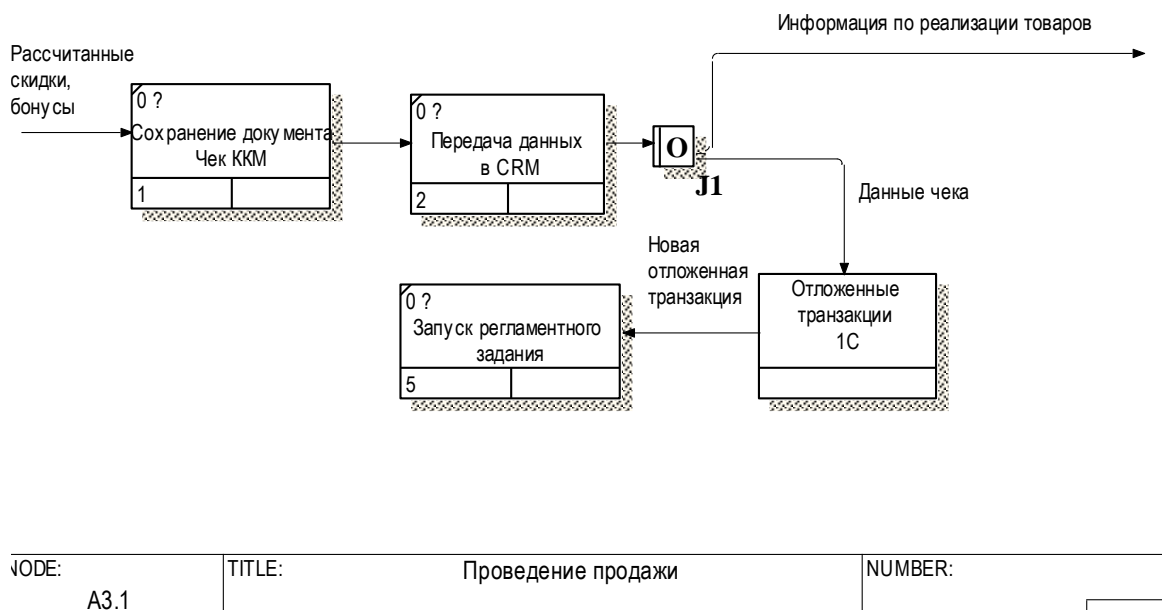


Рисунок 1.15 – Декомпозиция «Проведение продажи»

В результате анализа потоков данных был определен механизм выполнения бизнес-процесса продажи товаров и выявлены этапы, требующие оптимизации или доработки:

- 1) использование модуля «Sail Play»;
- 2) проведение продажи.

Было определено, что основной механизм при работе магазина «Terranova», требующий отладки для корректной работы отдела продаж, – это система «1С:Предприятие» конфигурации «Розница» версии 1.0.

2 Исследовательская часть

Для внесения изменений в алгоритм работы отдела продаж магазина «Terranova», необходимо переопределить входящие, выходящие и управленческие потоки и механизмы. Данная операция необходима для определения процессов, которые требуется заменить или модернизировать.

Моделирование потоков данных «Как будет» основано на модели «Как есть» и разрабатывалось посредством программы BPWin. При моделировании использовались методологии IDEF0 и IDEF3.

На рисунке 2.1 представлена диаграмма процесса осуществления продаж, в которой показано замещение механизма CRM-системы «Sail Play» на «Juice Development».

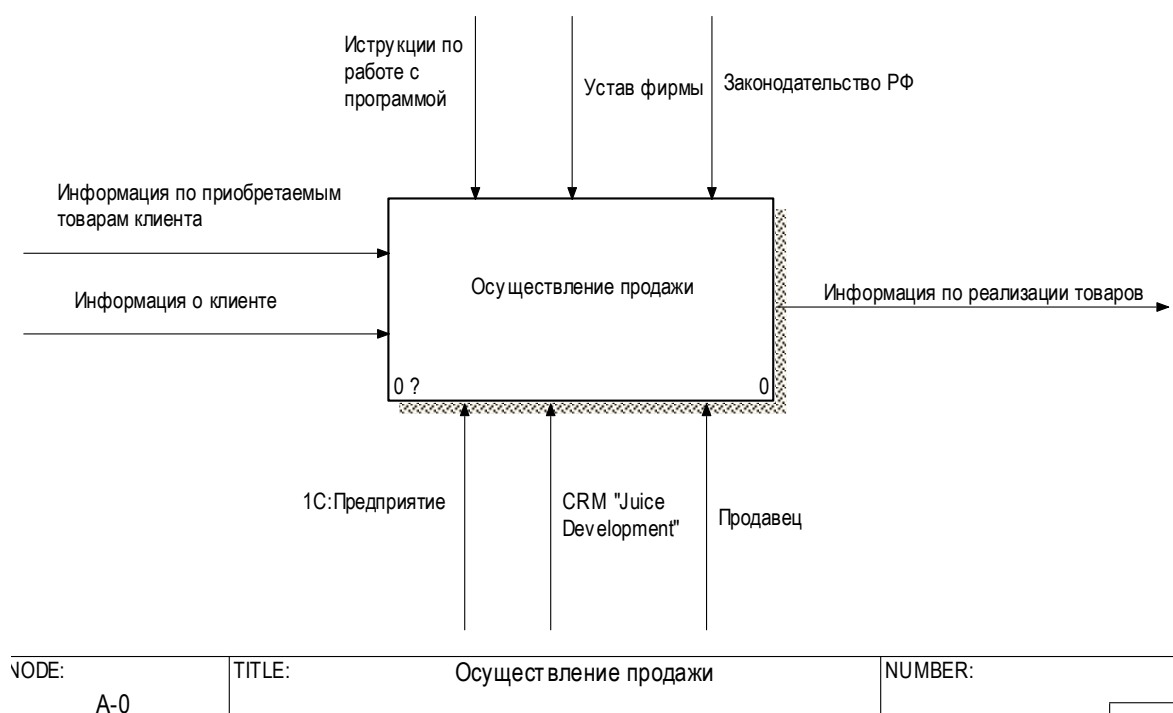


Рисунок 2.1 – Диаграмма «Осуществление продаж»

На рисунке 2.2 представлена основная диаграмма декомпозиции процесса осуществления продаж. На данной декомпозиции заменен блок «Использование модуля Sail Play» на «Использование модуля Juice Development».

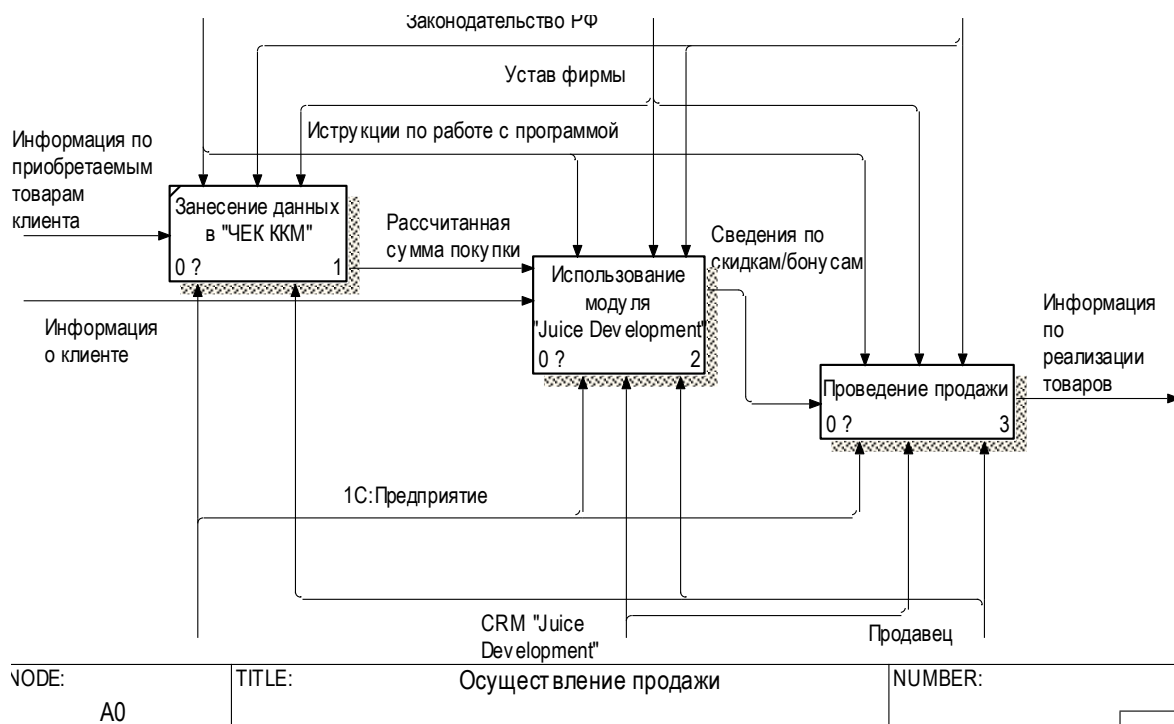


Рисунок 2.2 – Диаграмма декомпозиции «Осуществление продажи»

Рассмотрим представленную выше декомпозицию более подробно, с помощью декомпозиции второго уровня (рисунки 2.3-2.4), которые описывают какие процессы будут преобразованы.

На рисунке 2.3 были выполнены следующие модернизации:

- 1) заменен механизм «CRM Sail Play» на «CRM Juice Development»;
- 2) блок «Определение суммы скидки (списанные/начисленные бонусы)» разделен на два блока «Определение суммы скидки» с механизмом взаимодействия «Продавец» и «Расчет суммы начисленных бонусов» с механизмами «1С:Предприятие» и «CRM Juice Development»;
- 3) добавлен блок «Проверка корректности введенной информации» с механизмом «1С:Предприятие».

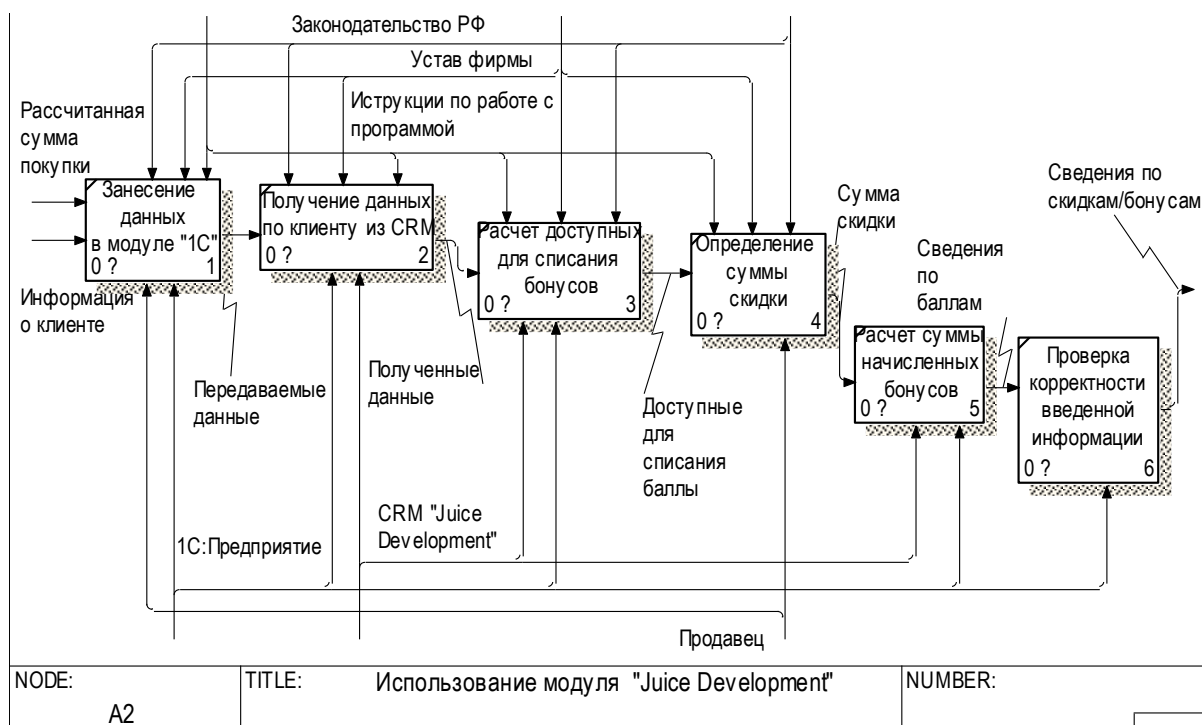


Рисунок 2.3 – Декомпозиция «Использование модуля Juice Development»

На рисунке 2.4 представлена декомпозиция процесса работы с данными клиента. Данная диаграмма выполнена в методологии IDEF3 и представляет следующие внесенные изменения:

- 1) добавлено разветвление объекта-ссылки «Отложенные транзакции» с помощью функционального элемента «Исключающее или», позволяющее выбрать один из двух входящих потоков к выполнению, со связями предшествования «Новая отложенная транзакция» и «Отложенная транзакция по расписанию»;
- 2) добавлена связь между процессами «Запуск регламентного задания» и «Передача данных в CRM»;
- 3) разветвлен функциональный механизм «Логическое или» связью предшествования «Данные по количеству попыток»;
- 4) заменена связь предшествования «Данные чека» на «Данные нового чека».

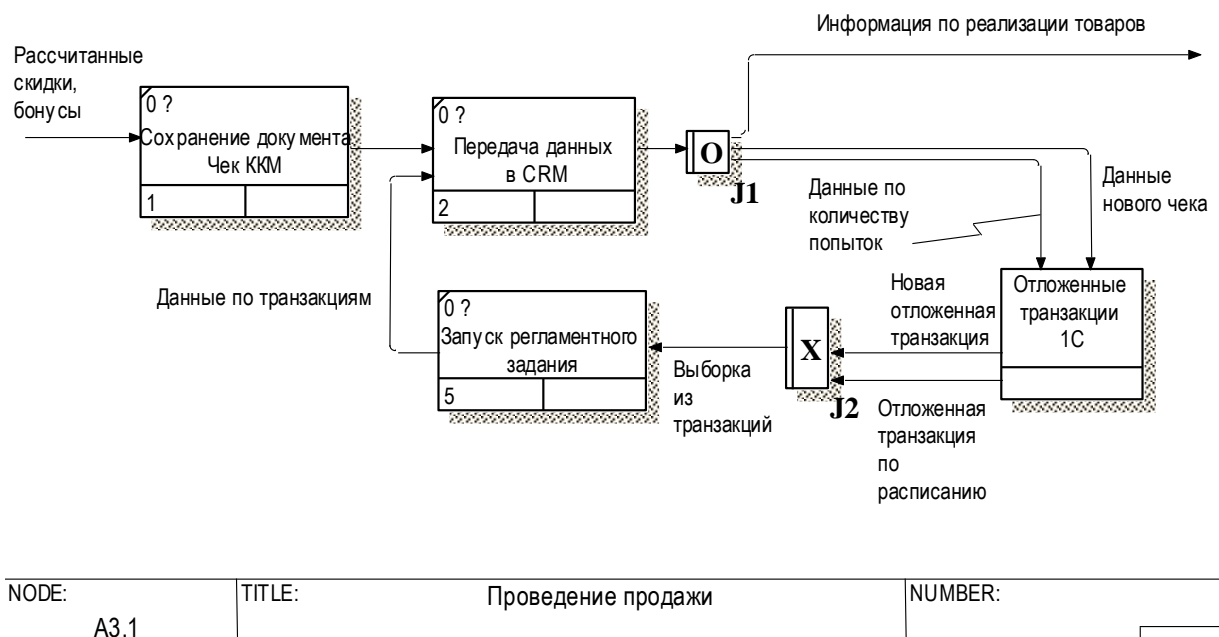


Рисунок 2.4 – Декомпозиция «Проведение продажи»

Рассмотрим модернизируемый механизм – конфигурацию «Розница» версии 1.0.

Конфигурация «Розница» предназначена для автоматизации бизнес-процессов магазинов, которые могут входить в распределенную розничную сеть торгового предприятия. Может использоваться для автоматизации магазинов с большим количеством рабочих мест, в том числе и в качестве кассовой программы.

В программе реализован многофирменный учет, при котором каждый склад магазина может быть отнесен к определенной организации. Для определенных торговых залов магазина может быть назначен учет ЕНВД (система налогообложения в виде единого налога на вмененный доход).

«1С:Предприятие» конфигурация Розница версия 1.0 автоматизирует типичные процессы розничного предприятия.

Данное программное решение в типовом варианте имеет функциональные возможности в следующих областях деятельности организации:

- 1) управление:
 - а) нормативно-справочной информацией;

- б) маркетингом;
- в) запасами и закупками;
- г) складом;
- д) продажами;
- е) финансами;
- ж) персоналом.

2) взаимодействие:

а) ЕГАИС (единой государственной автоматизированной информационной системой) – механизм автоматизации государственного контроля за объемом производства и оборота этилового спирта, алкогольной и спиртосодержащей продукции;

б) внешнее оборудование;

в) программные продукты «1С:Предприятие».

Из вышеописанного функционала наблюдается, что прямого взаимодействия между веб-CRM и Розница 1.0 нет. Для интеграции данных систем организация ранее обращалась к фирме для выполнения доработок на стороне программного продукта «1С:Предприятие».

В связи с тем, что затраты на обслуживание CRM были весьма значительны и проблемы при работе сотрудников в доработанной базе Розница 1.0 периодически возникали, организация пришла к решению заменить веб-приложение на более дешевую версию «под-ключ», а также в настроить программу «1С:Предприятие» для работы данной системой.

Основными причинами необходимости смены алгоритма работы послужили переход на CRM-систему «Juice Development» и обнаружение недостатков и ошибок при взаимодействии CRM и «1С:Предприятие», таких как:

1) при отсутствии интернета данные заносятся в список «Отложенные транзакции» и не отправляются в «Sail Play». Для передачи данных требуется использовать специальную кнопку «Выгрузить отложенные транзакции»;

2) в отложенных транзакциях отсутствует счетчик неудачного количества попыток передачи чеков;

3) отсутствует возможность изменения процентов, по которым рассчитываются начисления и списания бонусов с продажи. Данный коэффициент задан статически в программном коде;

4) не существует проверка по количеству списываемых баллов для скидки по бонусной программе по стоимости товаров (максимально возможный процент от общей суммы бонусов).

На основании анализа вышеописанных проблемных ситуаций был сделан вывод о необходимости выполнения следующих действий:

1) преобразовать механизмы передачи в CRM-систему отложенных транзакций по расписанию, в зависимости от указанного количества неудачных попыток;

2) добавить требуемые объекты для регулярного запуска операции по отправке данных.

3) изменить механизм расчета процентов для начисления/списания бонусов, путем их установки в CRM в соответствующем справочнике и дальнейшего получения в «1С:Предприятие».

Описание нового алгоритма, для решения имеющихся проблем и оптимизации обмена между базами, представлено ниже (рисунки 2.5-2.6).

Первая часть алгоритма изображена на рисунке 2.5. При открытии программы «1С:Предприятие» конфигурации «Розница» 1.0 запускается обработка «Меню РМК (рабочего места кассира)», которая проверяет наличие сети интернет и подключен ли фискальный регистратор. При любой из возникших проблем пользователю выводится сообщение об ошибке, но при отсутствии связи с фискальным регистратором обработка не позволяет отражать операции продажи.

В рабочем месте кассира пользователь-продавец должен инициализировать себя для отражения реализации товаров.

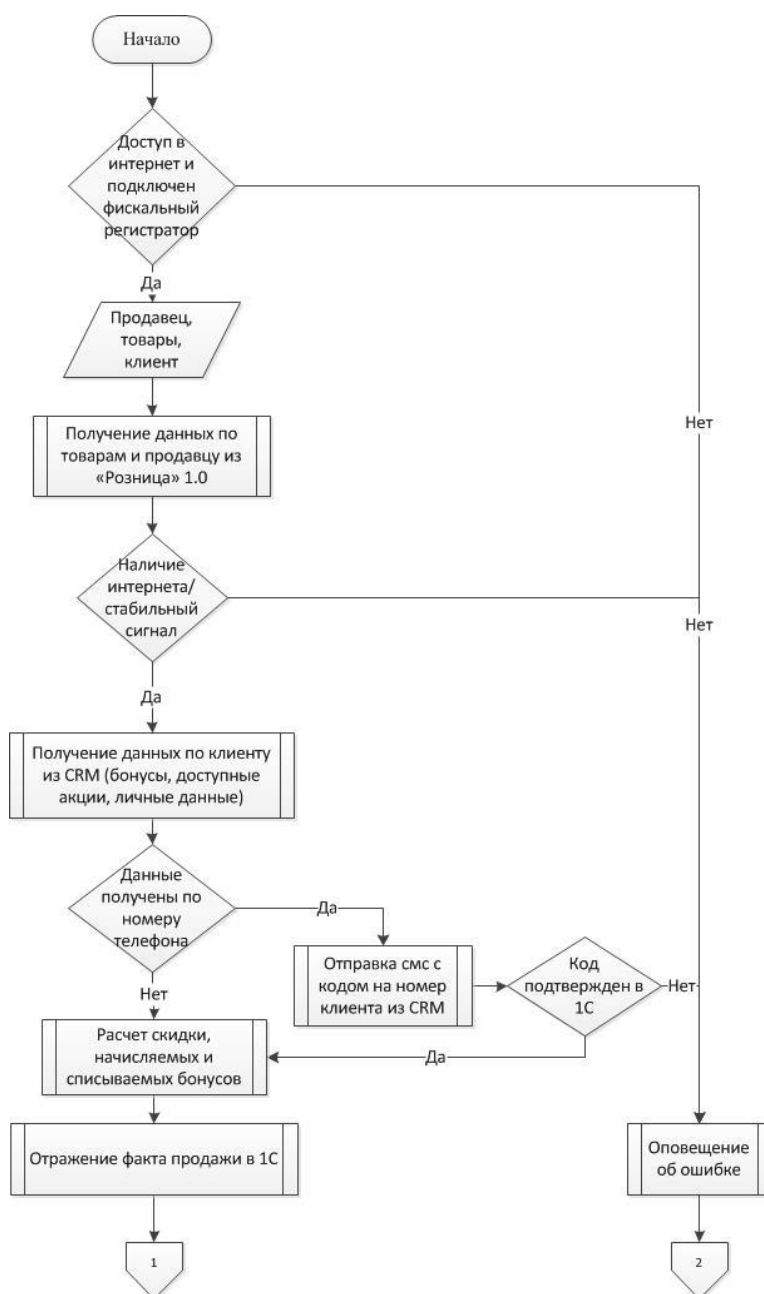


Рисунок 2.5 – Новый алгоритм продаж (1)

Когда клиент определился с покупками, он передает товары для занесения их в документ «Чек ККМ» в программе «1С:Предприятие». Пользователь отражает эти данные и получает рассчитанную предварительную сумму покупки. Если у клиента имеется накопительная бонусная карта, тогда продавец просит покупателя предоставить карту или по номеру телефона выполнить списание/начисление бонусов.

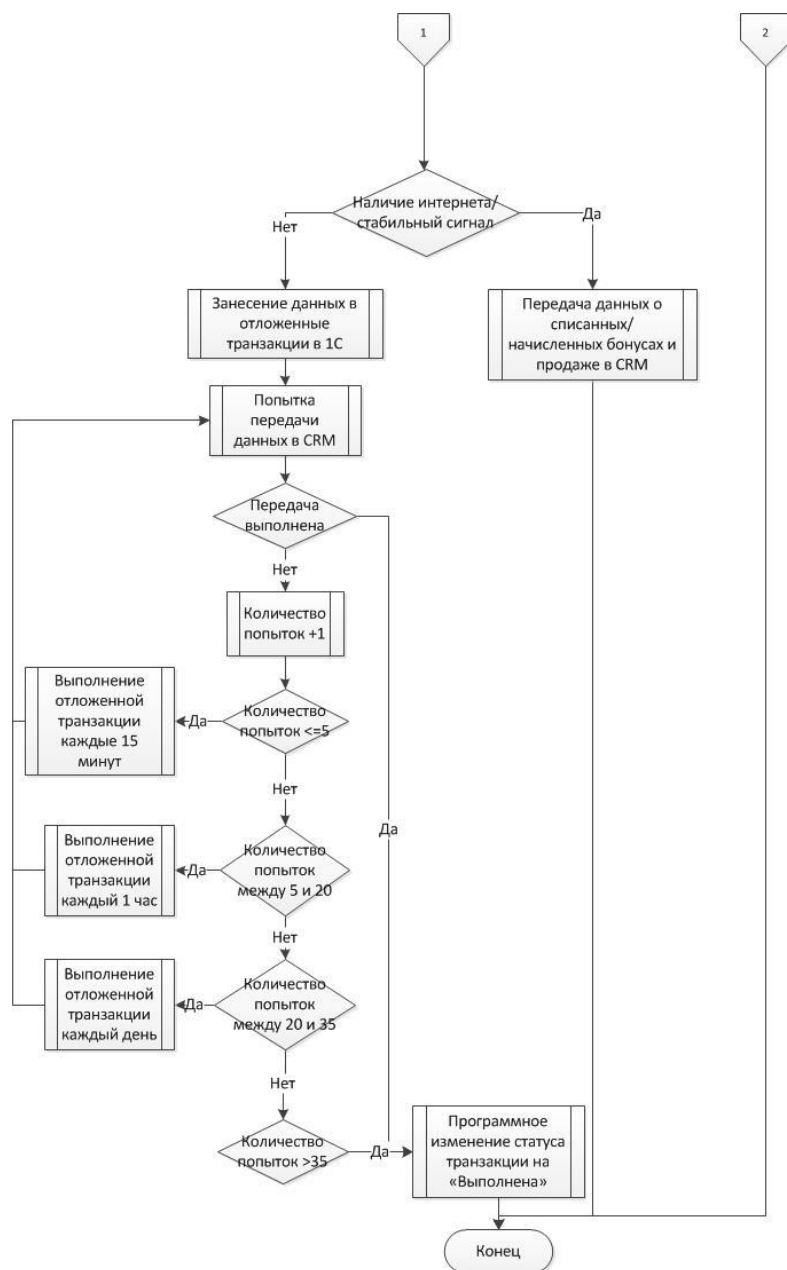


Рисунок 2.6 – Новый алгоритм продаж (2)

Вторая часть алгоритма представлена на рисунке 2.6. На данном этапе программа определяет наличие стабильного доступа к сети интернет. Если связь отсутствует, тогда «1С:Предприятие» оповещает об этом. В таком случае пользователь имеет возможность отложить чек до появления сети. В обратном случае программа отправляет запрос по введенным данным клиента в CRM-систему и получает ответ, содержащий данные покупателя. В «Рознице» производится расчет доступных для списания бонусных баллов и, после ввода количества списываемых баллов продавцом, рассчитывается количество

начисляемых баллов.

При выполнении операции списания/начисления баллов «1С:Предприятие» проверяет, запрос был отправлен с данными номера телефона или номером карты.

В первом случае пользователю направляется смс с кодом подтверждения. При несовпадении кода из CRM и введенного кода в «1С:Предприятие» выводится сообщение и операция по списанию/начислению бонусов не проводится. Во втором случае, данные по списанию и начислению бонусов заносятся в переменные.

Данные по списанным бонусам отражаются как скидка на всю покупку.

Продавец проводит документ продажи и подтверждает оплату наличными или картой. После чего программа «1С:Предприятие» выполняет попытку передачи и записи новых данных о продаже в CRM «Juice Development».

Если передача прошла без возникновения ошибок, тогда на этом акт продажи завершается. В противном случае программа оповещает об ошибке и заносит данные по чеку ККМ в список «Отложенные транзакции» с количеством попыток равным единице.

Отложенные транзакции разделяются по количеству попыток. В зависимости от чего они пытаются выполнить передачу данных в CRM каждые 15 минут, 1 час, 1 день, записываются, как выполненные. С каждой неудачной попыткой передачи данных количество попыток увеличивается на единицу.

Опишем инструменты для приведения в действие вышеописанного алгоритма.

Обмен данными производится в одностороннем порядке. Использование данного вида интеграции позволяет избежать ошибок, связанных с синхронизацией и сопоставлением информации, находящейся в двух различных базах. Также односторонний обмен подразумевает ведение учета по ключевым показателям в одной (основной) базе данных и передачи для анализа во вторую.

Интеграция между CRM «Juice Development» и «1С:Предприятие» конфигурации «Розница 1.0» будет осуществляться посредством использования механизма HTTP-запроса. Методы GET и POST будут применяться для передачи данных из 1С в веб-сервис CRM, который выполнен средствами API (англ. application programming interface, программный интерфейс приложения, интерфейс прикладного программирования) – набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах.

Метод GET предназначен для получения данных из указанного ресурса. Метод POST – для передачи данных, подлежащие обработке, на указанный ресурс.

Основное отличие данных методов при использовании в программе «1С:Предприятие» – это объявление заголовков и тела запроса.

Общий принцип выполнения HTTP-запросов выглядит следующим образом:

- 1) создаем класс HTTP-соединения с заданным адресом сервера;
- 2) определяем метод взаимодействия с CRM;
- 3) при использовании Get-запроса обращаемся к методу «Получить()», передавая параметр с данными адрес на ранее указанном сервере. На этом работа с методом Get завершена, сразу переходим к пункту номер 6.

При обращении к методу POST требуется выполнить действия, описанные в пунктах номер 4-7.

- 1) прописываем тело запроса;
- 2) добавляем требуемые параметры в запрос;
- 3) передаем запрос серверу;
- 4) получаем результат.

Структура запросов представлена на рисунке 2.7.

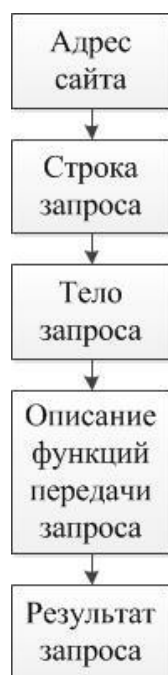


Рисунок 2.7 – Структура HTTP-запросов

Адрес сайта – ссылка на ресурс, с которым будут производиться взаимодействия.

Срока запроса представляет собой набор параметров, передаваемых строкой. Вид передаваемых параметров определяется в веб-сервисе.

Тело запроса представляет собой дополнительные данные, посылаемые серверу, т.е. требуемые параметры для подключения к сайту, такие как тип подключения, кодировка и т.д.

Описание функций передачи запроса определяется возможностями платформы «1С:Предприятие». При создании вышеописанного обмена использовалась платформа версии 8.2. Это обуславливается тем, что на торговых точках, которых у данного индивидуального предпринимателя более 10. При изменении версии платформы потребовалось приобретение новых лицензий и индивидуальная переустановка на каждом рабочем месте. Индивидуальным предпринимателем было принято решение отказаться от обновления платформы.

Результат запроса представляет собой ответ сайта на выполняемый запрос. Данный ответ состоит из следующего:

- 1) статус результата выполнения («ок», «error»)

- 2) код, соответствующий статусу и результату;
- 3) передаваемые параметры.

Таким образом, будут решены основные задачи организации:

- 1) оптимизация механизма реализации товаров средствами «1С:Предприятие»;
- 2) исправление ошибок в работе, мешающих оперативности сотрудников отдела продаж;
- 3) интеграция с веб-CRM «Juice Development».

3 Практическая часть

3.1 Программная реализация

В данной диссертационной работе производилась доработка типового функционала конфигурации «Розница» версии 1.0. Модернизирование функциональных возможностей производилось непосредственно над конфигурацией, поставляемой поставщиком. Механизм расширений конфигурации не использовался. Это связано с тем, что в платформе версии 8.2 он отсутствует.

Расширение конфигурации – механизм, являющий собой структуру типовой конфигурации, с возможностью внесения изменений без непосредственного редактирования объектов конфигурации поставщика.

Изменения производились над следующими объектами базы 1С:

- 1) общими модулями;
- 2) документами;
- 3) регистрами;
- 4) обработками;
- 5) регламентными заданиями.

В базу данных CRM «Juice Development» будут заноситься данные по всем магазинам, принадлежащим ИП Свечкарь С.А.. Для разграничения данных по реализациям товаров на точки продажи, параметры, которые передают значения «Код магазина» и «Ключ магазина», добавлены с реквизиты справочника и в константы (рисунок 3.1). Обращение к данным сведениям происходит в модуле сеанса конфигурации Розница версии 1.0.

Рисунок 3.1 – Параметры магазина

Рассмотрим основной доработанный функционал в разрезе вышеописанных программных единиц.

Основной проблемой при обмене данными в магазине является многократное обращение к отложенным транзакциям без упорядочивания и использования лишних процедур, что замедляет процесс передачи и обработки данных.

Для хранения данных по отложенным транзакциям предназначен одноименный непериодический независимый регистр сведений. В состав данного объекта включены:

- 1) измерения – документ (чек ККМ, чек ККМ на возврат, возврат товаров и услуг), вид операции (продажа товаров, возврат товаров);
- 2) ресурсы – транзакция завершена (истина, ложь).

В соответствии с алгоритмом, представленным во второй главе диссертационной работы, в данный объект был добавлен счетчик попыток в ресурсы (рисунок 3.2).

В форме объекта присутствует функция ручной выгрузку всех неудачных транзакций, реализованная по нажатию на кнопку «Выгрузить отложенные транзакции».

Период	Чек ККМ	Вид операции	Транзакция завершена	Число попыток
28.04.2018 19:44:28	Чек ККМ 0000007166 от 28.04.2018 19:44:15	Продажа	✓	
28.04.2018 19:56:52	Чек ККМ 0000007167 от 28.04.2018 20:02:03	Возврат	✓	
28.04.2018 19:57:58	Чек ККМ 0000007168 от 28.04.2018 20:02:04	Продажа	✓	
28.04.2018 20:13:01	Чек ККМ 0000007169 от 28.04.2018 20:16:51	Продажа	✓	
28.04.2018 20:23:02	Чек ККМ 0000007170 от 28.04.2018 20:26:00	Продажа	✓	
28.04.2018 20:34:39	Чек ККМ ATR00019114 от 28.04.2018 19:32:06	Продажа	✓	
28.04.2018 20:34:40	Чек ККМ ATR00019115 от 28.04.2018 19:36:31	Продажа	✓	
28.04.2018 20:34:40	Чек ККМ ATR00019116 от 28.04.2018 19:43:31	Продажа	✓	
28.04.2018 20:34:40	Чек ККМ ATR00019117 от 28.04.2018 19:47:24	Продажа	✓	
28.04.2018 20:34:41	Чек ККМ ATR00019118 от 28.04.2018 19:50:49	Продажа	✓	
28.04.2018 20:34:41	Чек ККМ ATR00019119 от 28.04.2018 19:55:19	Продажа	✓	
28.04.2018 20:34:41	Чек ККМ ATR00019120 от 28.04.2018 19:56:09	Продажа	✓	
28.04.2018 20:34:42	Чек ККМ ATR00019121 от 28.04.2018 19:57:17	Продажа	✓	
28.04.2018 20:34:42	Чек ККМ ATR00019122 от 28.04.2018 19:58:22	Продажа	✓	
28.04.2018 20:34:42	Чек ККМ ATR00019123 от 28.04.2018 19:59:20	Продажа	✓	
28.04.2018 20:34:43	Чек ККМ ATR00019124 от 28.04.2018 20:00:29	Продажа	✓	
28.04.2018 20:34:43	Чек ККМ ATR00019125 от 28.04.2018 20:01:44	Продажа	✓	
28.04.2018 20:34:43	Чек ККМ ATR00019126 от 28.04.2018 20:02:37	Продажа	✓	
28.04.2018 20:34:43	Чек ККМ ATR00019127 от 28.04.2018 20:04:46	Продажа	✓	
28.04.2018 20:34:44	Чек ККМ ATR00019128 от 28.04.2018 20:07:01	Продажа	✓	
28.04.2018 20:34:44	Чек ККМ ATR00019129 от 28.04.2018 20:17:05	Продажа	✓	
28.04.2018 20:34:44	Чек ККМ ATR00019130 от 28.04.2018 20:20:14	Продажа	✓	
28.04.2018 20:34:45	Чек ККМ ATR00019131 от 28.04.2018 20:23:15	Продажа	✓	
28.04.2018 21:02:06	Чек ККМ ATR00019132 от 28.04.2018 20:53:35	Продажа	✓	
28.04.2018 21:02:07	Чек ККМ ATR00019133 от 28.04.2018 20:54:21	Продажа	✓	1
16.05.2018 22:25:36	Чек ККМ 0000007176 от 16.05.2018 22:25:33	Продажа	✓	1

Рисунок 3.2 – Регистр сведений «Отложенные транзакции»

При использовании файловой версии базы данных программы «1С:Предприятие» конфигурации «Розница» версии 1.0 автоматический запуск регламентных заданий невозможен. В связи с чем, изменения по передаче отложенных транзакций выполнялись с использованием модуля конфигурации и общего модуля. Для исключения ситуаций частой передачи одного и того же объекта с ошибкой был добавлен общий модуль «Отложенные транзакции», где прописан алгоритм отбора транзакций для обмена по счетчику попыток.

Для запуска данной процедуры было инициировано обращение к функциям, передающим значения по фильтрации транзакций и временному интервалу запуска, рассчитанного в секундах. Данным механизм реализован в модуле обычного приложения. Запуск задач по расписанию происходит автоматически после загрузки программы «1С:Предприятие» и продолжается на протяжении работы программы вплоть до ее закрытия.

Так же реализована отправки данных по неудачным транзакциям после проведения чека, если его отправка не была выполнена сразу после создания. Данная операция осуществляется один раз и присваивает количеству попыток значение равное единице.

После преодоления коллизий при передаче данных в CRM была выполнена полная замена существующих дописанных функций на новые, соответствующие методу API данной системы. Отдельная работа была

произведена над алгоритмами, обрабатывающими ошибки. Основные ошибки были добавлены в механизм продажи и работы с клиентом. Осуществлен их вывод пользователю в доступной форме (рисунок 3.3-3.4).

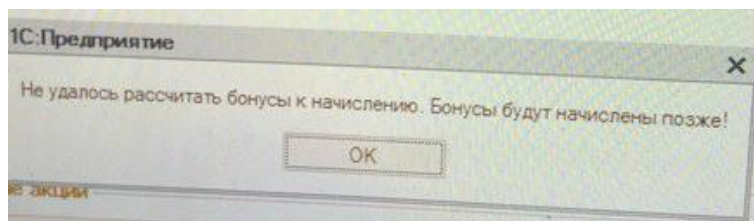


Рисунок 3.3 – Пример ошибки при регистрации продажи

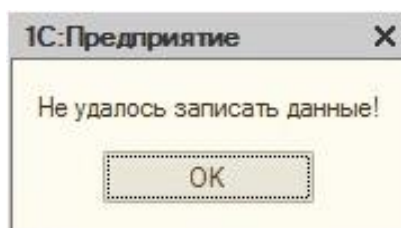


Рисунок 3.4 – Пример ошибки при регистрации клиента

Рассмотрим изменения, внесенные в документ «Чек ККМ».

Для отражения операций по реализации товаров необходимо зайти в программу от пользователя «Кассир». В окне «Выбор действия рабочего места кассира» необходимо открыть смену, нажав на кнопку «Открытие смены».

Для работы с данными по бонусной программе и индивидуальными данными клиентов в форму документа «Чек ККМ», являющегося основным рабочим местом сотрудников, была добавлена кнопка «Juice Develop», расчет скидки по клиенту и одна попытка передачи текущей отложенной транзакции после неудачной отправки данных по регистрируемому чеку.

Перед занесением товаров, приобретаемых клиентом, в список реализаций сотруднику необходимо пройти идентификацию в рабочем месте кассира (рисунок 3.5). Данная операция выполняется по нажатию на кнопку «Продавец» и выборе соответствующего сотрудника из списка. При отсутствии данных о продавце программа выведет сообщение о соответствующей ошибке и не позволяет отразить факт продажи товаров клиенту.

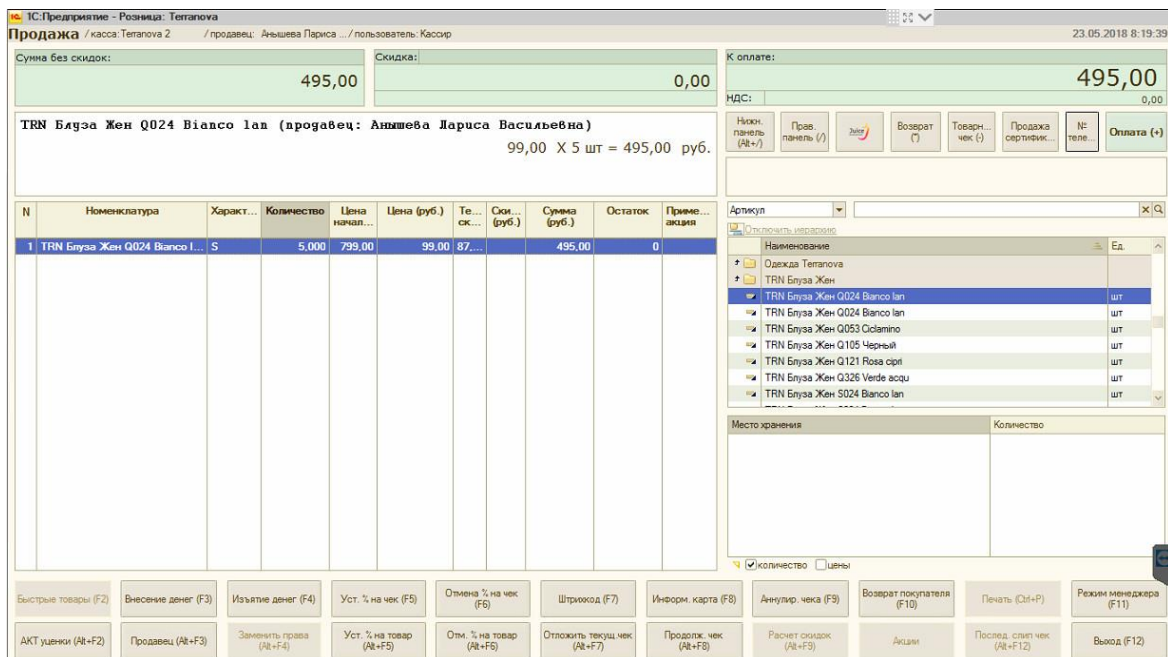


Рисунок 3.5 – Рабочее место кассира

После регистрации продавца в системе становится доступна возможность отражения продаж. Для этого необходимо выбрать товары из списка номенклатуры и указать соответствующее количество и цену, если на данный товар не было оформлено документов установки цен.

Как только список соответствует выбранным позициям клиента, если покупателю была ранее выдана бонусная карта с баллами, можно отразить скидку в соответствии с процентом, указанным в бонусной программе CRM (рисунок 3.6).

ОТДЕЛ	ID ОТДЕЛА	ТОКЕН	НАЧИСЛЕНИЯ	МАКС. СКИДКА	API АКТИВНО
Paolo Conte BCM ТЦ СитиМолл, Белгород	853705	*****LXI	7%	30%	✓
Paolo Conte BMG ТЦ МегаГринн, Белгород	698459	*****d22	7%	30%	✓
Respect Белгород	426436	*****EGA	7%	30%	✓
Sportactive ТЦ Центральный, Белгород	968759	*****VLN	7%	30%	✓
Stin Универмаг Белгород	130955	*****Ndw	7%	20%	✓
Stin BMB ТЦ Модный бульвар, Белгород	410790	*****bAf	7%	20%	✓
Terranova Народный Бульвар, Белгород	140254	*****jo1	0%	5%	✓

Рисунок 3.6 – Данные по бонусной программе в CRM

Далее работа ведется с модулем «Juice Development». Рассмотрим изменения, внесенные в обработку по работе с контрагентами.

Данный модуль предназначен для регистрации личных данных клиента, сведений о бонусной карте и количества доступных баллов, позволяющих оформить скидку при покупке. Информация по покупателю хранится только на стороне CRM.

Редактирование данных клиента

Номер карты:

Номер телефона:

Статус карты:

Фамилия:

Имя:

Почта:

Пол:

Дата рождения:

Рисунок 3.7 – Форма работы с данными клиента

Основные данные, требуемые для передачи в веб-систему, представлены на рисунке 3.7. Данная форма предназначена для регистрации, получения и изменения сведений по клиенту.

Основная форма, предоставляющая возможность работы с данными, представлена на рисунке 3.8.

The screenshot shows a software window titled "Обработка sail play" with a close button (X) in the top right corner. The form is divided into several sections:

- Авторизация:** Contains input fields for "Сканирование карты:" (9999200041535), "Номер телефона:" (+7 (910) 745-50-55), and "Код подтверждения:" (0). There are buttons for "ШК" and "OK".
- Buttons:** A large button labeled "Пройти авторизацию" is centered below the authorization fields.
- Client Information:** Includes "Имя:" (Ольга) and "Отчество:" fields, with a "Редактировать клиента" button.
- Возможные акции:** A table with columns "N" and "Акция" is currently empty.
- Применяемые акции:** Another table with columns "N" and "Акция" is also empty.
- Buttons:** A button labeled "Рассчитать акции" is centered below the tables.
- Оплата бонусами:** Contains input fields for "Сумма чека:" (495,00), "Сумма бонусов:" (5,00), "Сумма к оплате:" (495,00), and "Бонусов, начислено за покупку:" (0,00).
- Buttons:** A large orange button labeled "Списать бонусы" is centered below the payment section.
- Footer:** A "Закрыть" button is located in the bottom right corner.

Рисунок 3.8 – Форма обработки

В форме работы с данными клиента были внесены следующие изменения:

- 1) запрет редактирования номере карты уже занесенных данных по клиенту;
- 2) заблокировано поле «Статус карты» в зависимости от наличия маркетинговых акций;
- 3) заменено поле «Отчество» на поле «Почта», добавлен соответствующий реквизит в форму.

Работает данный модуль следующим образом.

Получить сведения по клиенту можно по предоставленному им номеру карты или телефона. Для выполнения данной операции необходимо ввести соответствующие сведения в соответствующее поле на форме (рисунок 3.7), после чего нажать на кнопку «Пройти авторизацию».

При авторизации по номеру телефона происходит дополнительное действие – отправка смс сообщения на указанный номер с кодом подтверждения, сгенерированным и сохраненным в таблицу отправленных сообщений в системе CRM «Juice Development». Программный продукт «1С:Предприятие» регистрирует данное действие с помощью окна оповещения (рисунок 3.9). После чего программой «Розница» отправляется запрос для получения сведений по коду и производится сравнение соответствия введенного в форму модуля «Juice Development» по нажатию на кнопку «ОК», расположенную рядом с полем ввода данных по коду. Если сведения совпали, продавец может продолжить оформлять продажу, в обратном случае продавец получит оповещение о различии в кодах (рисунок 3.11). При несовпадении программа задает пользователю вопрос о необходимости выполнения повторной отправки данных из CRM (рисунок 3.10). Если продавец отказывается от повторной операции, чек будет отложен. В дальнейшем у пользователя «1С:Предприятие» имеется возможность вернуться к оформлению продажи или удалить данные по отложенной реализации.

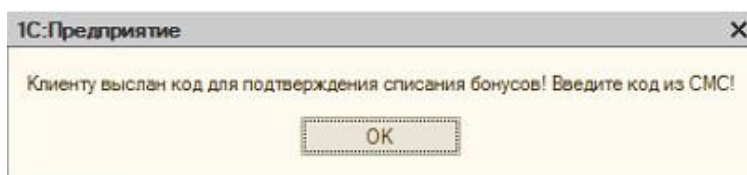


Рисунок 3.9 – Оповещение о выполненной операции передачи СМС-кода

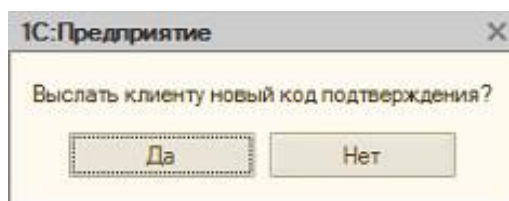


Рисунок 3.10 – Вопрос на повторение операции

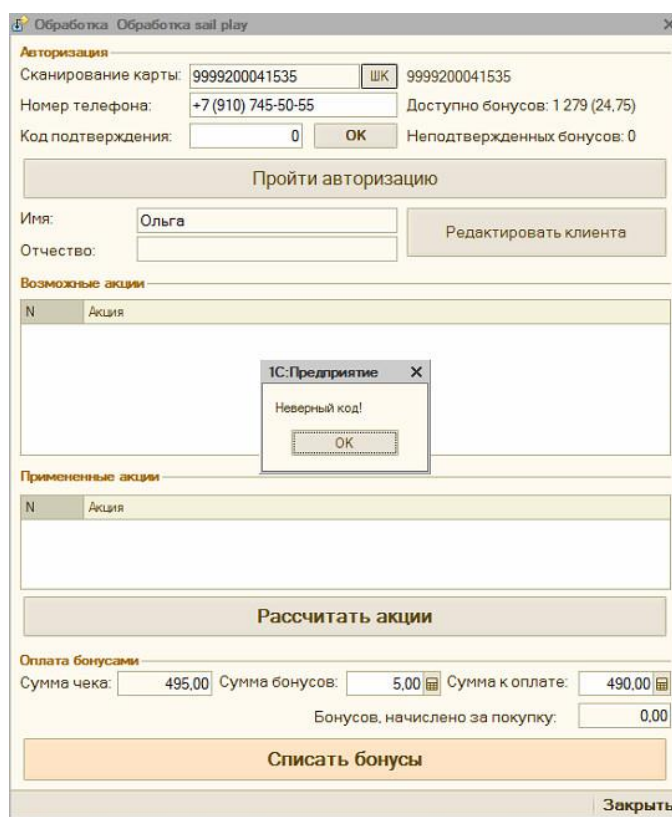


Рисунок 3.11 – Оповещение о несовпадении кода

При совпадении кода или получении данных по номеру бонусной карты программа направляет запрос с параметрами «Магазин», «Токен», «Код», «Номер телефона» или «Номер карты» в CRM «Juice Development». Результатом данной операции является статус выполненной транзакции и данные, соответствующие итогам процедуры. Если результат «ОК», тогда программа получает сведения о клиенте и его бонусах, иначе – код ошибки и информацию, по какой причине операция не была произведена.

Если данные по клиенту вернулись неточные или некорректные, в программе «1С:Предприятие» реализована возможность внести в них изменения. Для выполнения данной операции требуется нажать на кнопку «Редактировать клиента» и в открывшемся окне внести изменения. По окончании ввода верных данных требуется нажать на кнопку «Выполнить». При ошибке, возникающей в результате выполнения операции, продавец получает оповещение о неудачной транзакции и сведения, описывающие причину данной ситуации.

На рисунке 3.12 представлено отображение списка клиентов в модуле CRM «Juice Development», после передачи данных из «1С:Предприятие».

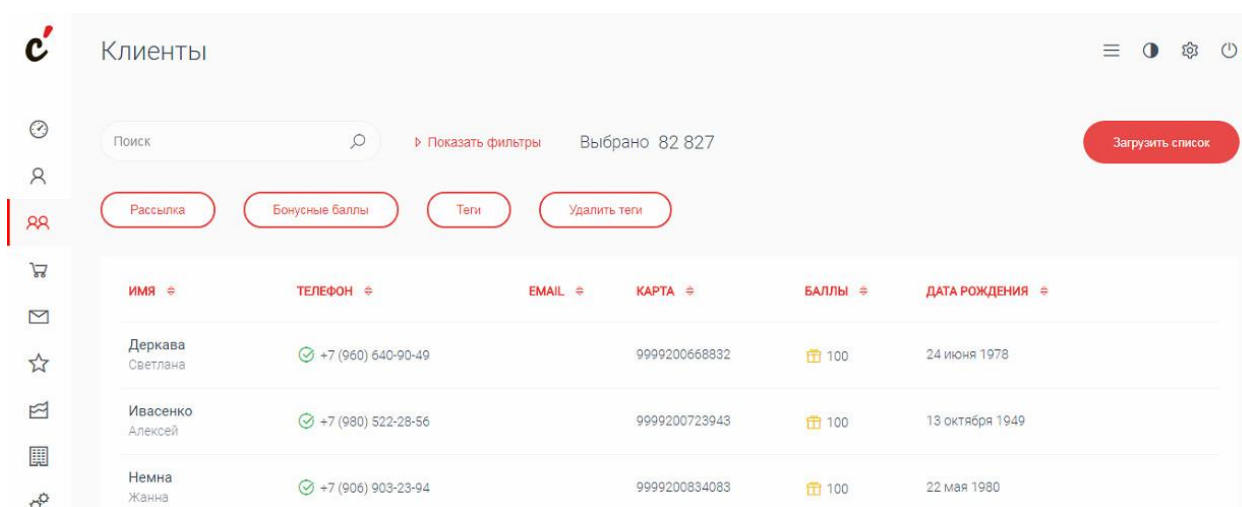


Рисунок 3.12 – Рабочее место кассира выбор товаров

После успешного получения данных о клиенте в модуле работы с покупателем отображаются его текущие баллы, где в скобках указано максимально возможное количество для списания (скидки на текущую стоимость покупки), рассчитанное по коэффициенту, введенному в бонусной программе в CRM.

Если в поле «Сумма бонусов» будут введены данные, превышающие значение максимально доступных для скидки, тогда пользователю выводится соответствующее оповещение и дальнейшая работа с чеком блокируется, а сумма списания обнуляется (рисунок 3.13).

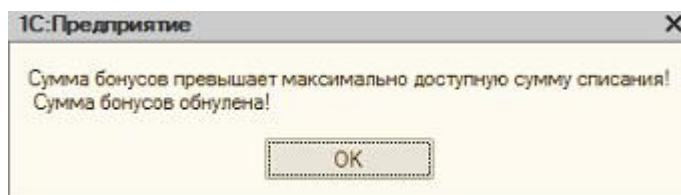


Рисунок 3.13 – Оповещение о некорректной сумме списания

При корректном вводе данных по списываемым бонусам происходит расчет начисляемых бонусов, в соответствии с бонусной программой магазина,

сохраненной в CRM. Это значение заносится в отдельную переменную и отправляется вместе с данными по чеку только при успешном осуществлении продажи.

Данные по списанным бонусам являются скидкой в денежном эквиваленте на общую стоимость покупки клиента и отражаются в модуле рабочего места кассира (рисунок 3.14). Так же происходит расчет процента скидки от общей суммы.

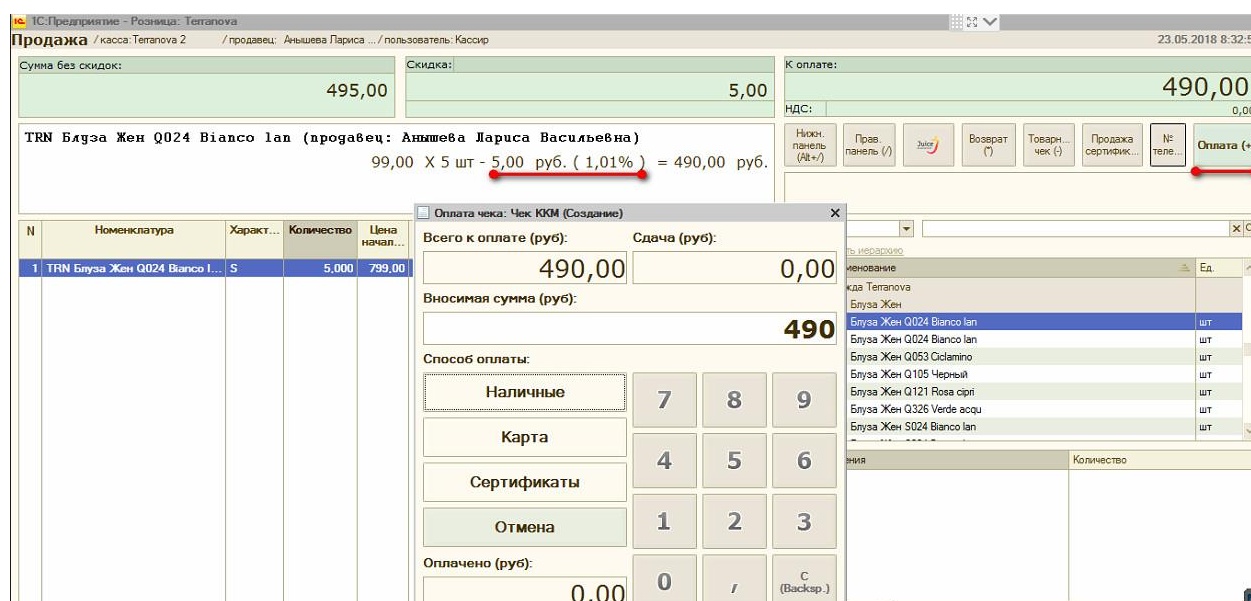


Рисунок 3.14 – Отражение скидки, оплата

Когда все данные по текущей реализации отражены в рабочем окне для фиксирования процесса оплаты необходимо нажать на соответствующую кнопку. В открывшемся окне «Оплата чека» выбирается способ взаиморасчетов с клиентом. Имеется три варианта отражения поступления платежа:

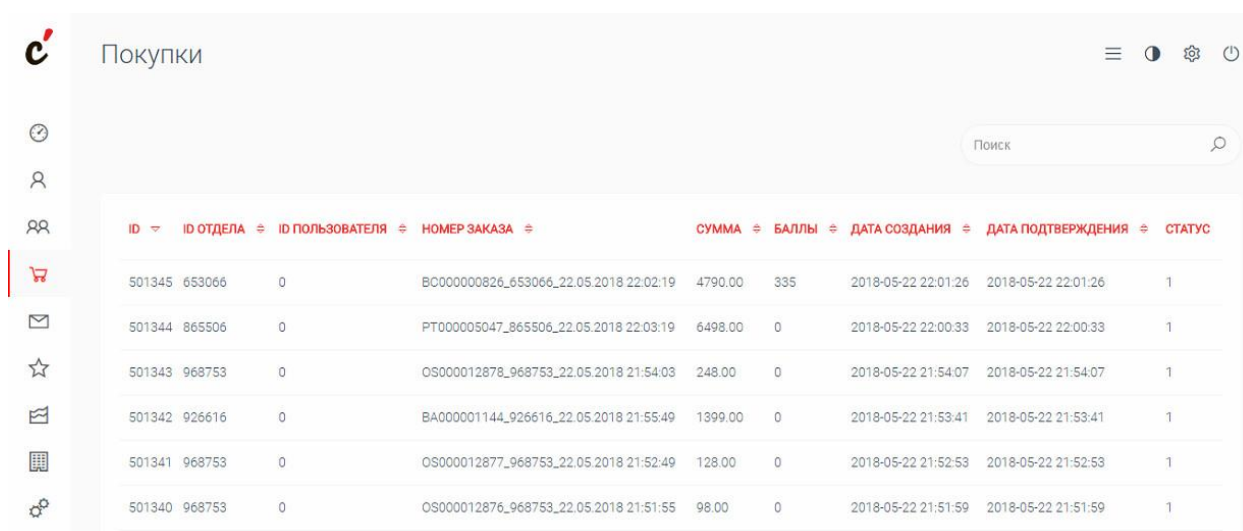
- 1) наличные;
- 2) карта;
- 3) сертификаты.

При выборе способа оплаты наличные или сертификаты результат взаиморасчетов вводится в поле «Вносимая сумма».

При оплате картой требуется произвести операцию через эквайринговый

терминал. Для выполнения данного действия в «1С:Предприятие» в модуле «Оплата чека», нажимается кнопка «Карта». Клиент вставляет карту в эквайринговый терминал. Происходит списание денежных средств с банковского счета покупателя, в «1С:Предприятие» передаются данные о номере карты и успешно выполненной операции. При возникновении ошибки или неудачной транзакции программный продукт оповещает об этом продавца в соответствующем окне.

При успешно выполненной операции происходит печать чека и в поле «Оплачено (руб.)» отображается сумма поступившего платежа. Все данные, представленные на рисунке 3.14, сохраняются в «1С:Предприятие» в документе «Чек ККМ». После сохранения в информационной базе сведений по операции, происходит процесс передачи информации в CRM «Juice Development». На рисунке 3.15 представлено окно «Покупки», в котором отражаются данные по реализации товаров.



ID	ID ОТДЕЛА	ID ПОЛЬЗОВАТЕЛЯ	НОМЕР ЗАКАЗА	СУММА	БАЛЛЫ	ДАТА СОЗДАНИЯ	ДАТА ПОДТВЕРЖДЕНИЯ	СТАТУС
501345	653066	0	BC000000826_653066_22.05.2018 22:02:19	4790.00	335	2018-05-22 22:01:26	2018-05-22 22:01:26	1
501344	865506	0	PT000005047_865506_22.05.2018 22:03:19	6498.00	0	2018-05-22 22:00:33	2018-05-22 22:00:33	1
501343	968753	0	OS000012878_968753_22.05.2018 21:54:03	248.00	0	2018-05-22 21:54:07	2018-05-22 21:54:07	1
501342	926616	0	BA000001144_926616_22.05.2018 21:55:49	1399.00	0	2018-05-22 21:53:41	2018-05-22 21:53:41	1
501341	968753	0	OS000012877_968753_22.05.2018 21:52:49	128.00	0	2018-05-22 21:52:53	2018-05-22 21:52:53	1
501340	968753	0	OS000012876_968753_22.05.2018 21:51:55	98.00	0	2018-05-22 21:51:59	2018-05-22 21:51:59	1

Рисунок 3.15 – Список покупок в CRM

Если при обмене данными возникла ошибка, выполняется запись в регистр сведений «Отложенный транзакции» с соответствующим видом операции и количеством попыток равным единице.

В дальнейшем отложенные транзакции отправляются в соответствии с

количеством tentативы, указанной для данной строки. Отправка данных происходит по следующему порядку:

- 1) при количестве попыток менее или равно 5 отправка данных происходит каждые пятнадцать минут;
- 2) для более 5 и менее или равно 20 каждые тридцать минут;
- 3) от 21 до 35 каждый день;
- 4) более 35 транзакции перестают отправляться. Такие операции необходимо добавлять вручную в систему CRM.

Для оформления возвратов товаров также используется рабочее место кассира.

В системе определяется продавец по вышеописанному принципу. Далее необходимо выбрать «Чек ККМ», по которому была проведена реализация данного товара. Для выполнения данной операции пользователь нажимает на кнопку «Возврат». Из формы выбора «Чек ККМ» продавец выбирает по номеру чека и дате нужный. После чего данные переносятся в рабочее место кассира, и кнопка «Возврат» заменяется кнопкой «Продажа» (рисунок 3.16).

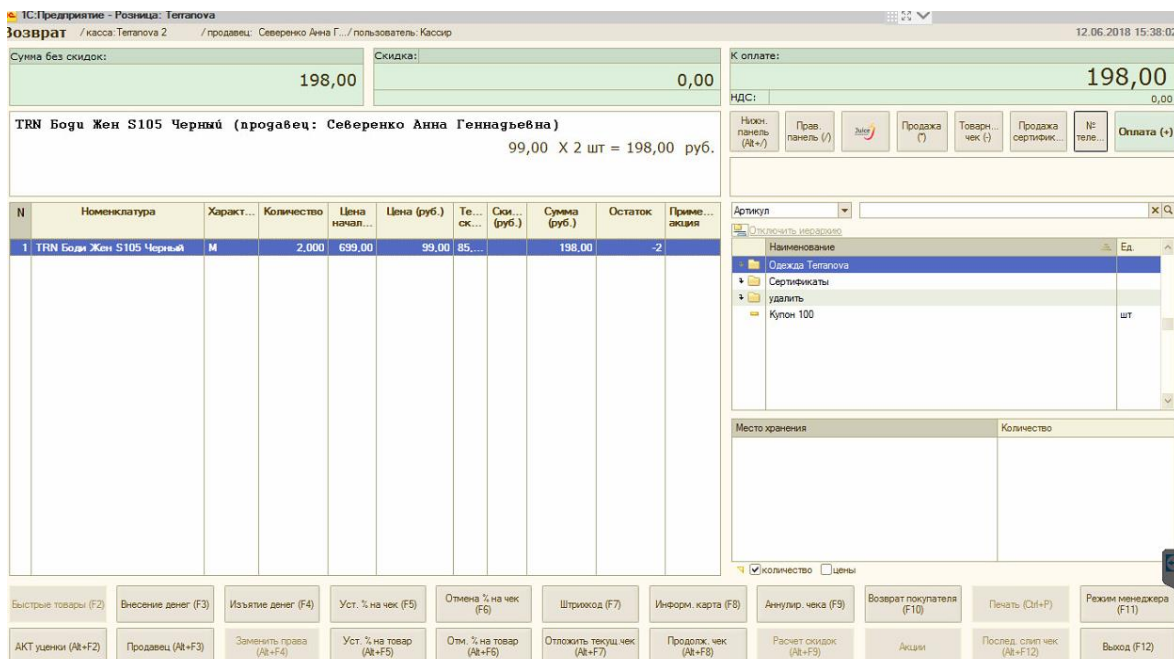


Рисунок 3.16 – Возврат товаров

При наличии списанных баллов, они отражаются, как процент ручной скидки. Возврат денежных средств производится по нажатию на кнопку «Оплата» (рисунок 3.17). Денежные средства возвращаются только в той форме, в которой были получены, т.е. наличные, сертификат или безналичные.

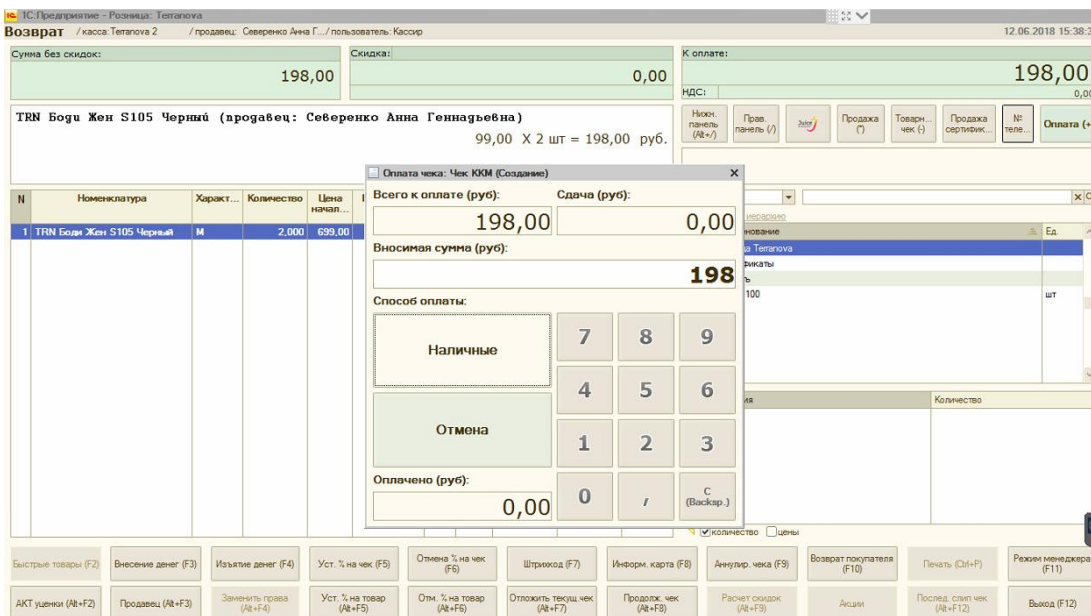


Рисунок 3.17 – Оплата возврата товаров

После проведения операции возврата товаров от клиента, данные передаются в CRM «Juice Development», изменяя статус проведенного чека с «проведен» на «возврат». Списанные и начисленные бонусные баллы также возвращаются в состояние, без учета данной продажи.

Таким образом, все операции отражаются в CRM корректно.

3.2 Эффективность

Целью данной диссертационной работы было повышение эффективности за счет устранения коллизий и простоев в работе. Данная операция была достигнута при помощи изменения алгоритма интеграции и замены CRM «Sail Play» на «Juice Development».

Работая с CRM «Sail Play» и до внесения изменений в систему

«1С:Предприятие» конфигурации Розница версии 1.0 коллизии возникали каждые 5 минут при передаче:

- 1) отложенных транзакций при плохом интернете;
- 2) данных по возврату по чеку, который был зафиксирован или не передан из-за ошибки или отсутствия интернета;
- 3) большого количества отложенных транзакций.

На рисунке 3.18 показан пример коллизий по одному документу из отложенной транзакции за 1 день.

id	method	store_department_id	params	code	date_add	status
7285	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	0	2018-04-28 09:57:51	1
8381	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 12:38:45	2
8478	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 12:48:45	2
8563	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 12:58:44	2
8692	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 13:08:52	2
8824	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 13:18:52	2
8981	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 13:28:52	2
9140	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 13:38:52	2
9326	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 13:50:17	2
9468	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 14:00:17	2
9738	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 14:26:57	2
9863	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 14:42:00	2
10072	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 14:55:48	2
10562	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 15:24:31	2
10751	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 15:35:28	2
10886	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 15:45:28	2
11037	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 15:55:24	2
11224	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 16:06:04	2
11425	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 16:16:03	2
11921	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 16:53:29	2
12104	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 17:03:29	2
12250	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 17:13:24	2
12395	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 17:23:10	2
12588	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 17:32:59	2
12967	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 17:53:02	2
13186	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 18:02:54	2
13458	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 18:14:23	2
13696	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 18:24:22	2
13906	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 18:34:22	2
14045	purchase.new	140254	{"phone": "...", "email": "...", "origin_user_id": "...", "order_...	35	2018-04-28 18:44:22	2

Рисунок 3.18 – Список коллизий в CRM за 1 день

Для решения проблемных ситуаций был выполнен отбор, который исключал старые операции за 2017 г. для передачи в новый CRM «Juice Development». После чего были произведены изменения по представленному во второй главе разработанному алгоритму и протестирован.

В результате внесенных изменений было снижено количество коллизий.

Результаты представлены на графике 3.19, где первые 15 дней мая – количество ошибок до внесения изменений в механизм работы магазина, вторые 15 дней – количество коллизий при применении нового алгоритма. Данные представлены за 1 месяц.

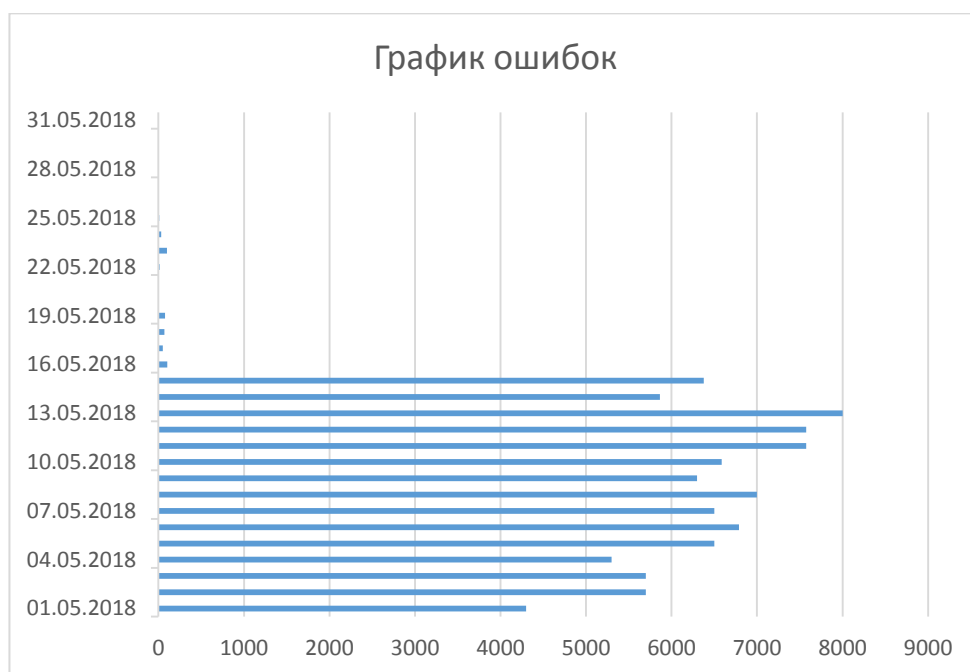


Рисунок 3.19 – График коллизий

Сумма коллизий за 15 дней до внедрения изменений составила 96097 обращений к серверу CRM. После исправления механизма работы Розницы версии 1.0 количество ошибок снизилось до 468 единиц за 15 дней.

При большом количестве ошибок при взаимодействии «1С:Предприятие» с «Sail Play» возникала фатальная перегрузка веб-системы. Это оказывало воздействие на Розницу в виде зависаний программы и простоев в работе отдела продаж. Количество простоев за 15 дней достигло 161 единицы.

На рисунке 3.20 показан график изменения количества зависаний, где до 16 мая – результат до внесения изменений, а после – с внедренным алгоритмом. Сведения представлены за 1 месяц, а количество простоев уменьшилось до 10 единиц.

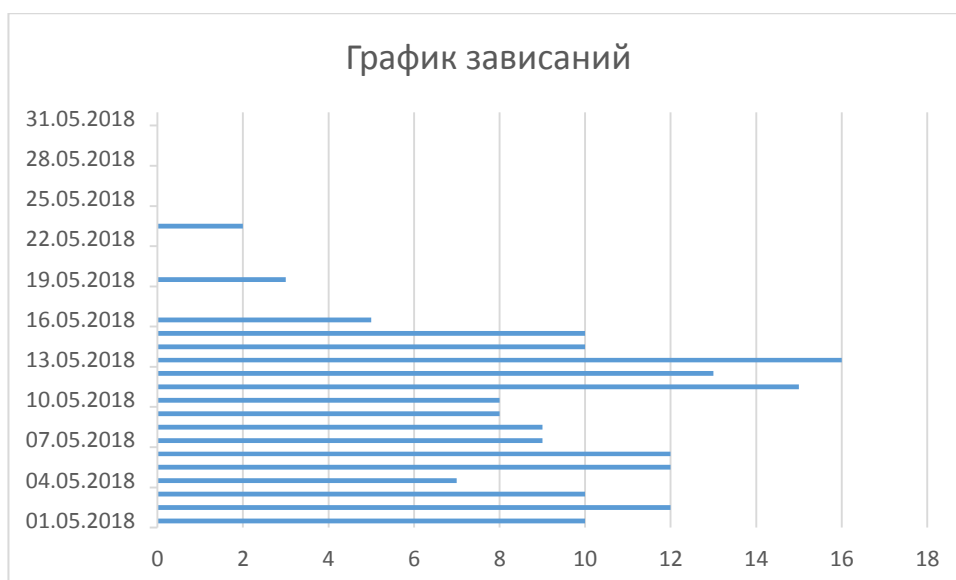


Рисунок 3.20 – График простоев

Данные, представленные на графиках наглядно показывают, насколько увеличилась эффективность работы отдела продаж магазина «Terranova».

Кроме вышесказанного уменьшились затраты магазина на обслуживание CRM-системы «Sail Play», которое до внедрения составляло 350 тысяч рублей в год. Веб-система «Juice Development» была разработана «под-ключ» и ее стоимость составила немногим менее двухгодовых затрат на обслуживание «Sail Play». При необходимости внесения доработок или поддержки со стороны организации «Juice Development», компании придется выделить дополнительные средства на данные действия.

Количество коллизий уменьшилось в 99 %, а количество простоев снизилось в 93 %, благодаря чему увеличилось быстродействие сотрудников и лояльность клиентов, что благотворительно сказывается на прибыли предприятия.

ЗАКЛЮЧЕНИЕ

При выполнении диссертационной работы была определена наиболее подходящая ERP-система для организации розничных продаж. После произведения сравнительного анализа методом МАИ был составлен новый алгоритм работы отдела продаж и осуществлено внедрение данного механизма в рабочую систему фирмы.

Итогом выполнения диссертационной стало увеличение эффективности работы отдела продаж магазина «Terranova». Данный результат был получен в следствии изменении алгоритма бизнес-процессов организации и замены CRM-системы на «Juice Development».

В диссертационной работе были выполнены следующие задачи:

- 1) изучены методы и подходы для управления бизнес-предприятием;
- 2) проанализированы существующие системы управления бизнес-предприятием, имеющие возможность интеграции со сторонними CRM-системами;
- 3) определен основной функционал и возможности синхронизации данных;
- 4) проанализирована работа организации, на основании чего была составлена модель «как есть» с помощью инструментальных средств представления информации;
- 5) создан и описан новый алгоритм выполнения бизнес-процесса розничной продажи;
- 6) разработана программа и протестирована;
- 7) рассчитана эффективность предложенного решения.

Цель была достигнута.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 7.32-2001. Отчет о научно-исследовательской работе. Структура и правила оформления [Текст]. – Взамен ГОСТ 7.32-91; Введ. 01.07.2002. – М: Стандартинформ, 2008. – 20 с. – (Система стандартов по информации, библиотечному и издательскому делу).

2. ГОСТ 2.105-95. ОБЩИЕ ТРЕБОВАНИЯ К ТЕКСТОВЫМ ДОКУМЕНТАМ [Текст]. – Взамен ГОСТ 2.105-79. ГОСТ 2.906-71; Введ. 01.06.1996. – Минск: Межгосударственный совет по стандартизации, метрологии и сертификации, 2007. – 30 с. – (Единая система конструкторской документации).

3. Положение о выпускных квалификационных работах дипломированного специалиста, бакалавра, по программам получения дополнительных квалификаций от 15 октября 2007 г. (с изменениями и дополнениями от 22.06.2009 г.) [Текст] – Белгород: Изд-во БелГУ, 2009. – 20 с.

4. ГОСТ 7.1-2003. БИБЛИОГРАФИЧЕСКАЯ ЗАПИСЬ. БИБЛИОГРАФИЧЕСКОЕ ОПИСАНИЕ. Общие требования и правила составления [Текст]. – Взамен ГОСТ 7.1-84, ГОСТ 7.16-79, ГОСТ 7.18-79, ГОСТ 7.34-81, ГОСТ 7.40-82; Введ. 01.07.2004. – М: Стандартинформ, 2006. – 52 с. – (Система стандартов по информации, библиотечному и издательскому делу).

5. ГОСТ 7.82-2001. БИБЛИОГРАФИЧЕСКАЯ ЗАПИСЬ. БИБЛИОГРАФИЧЕСКОЕ ОПИСАНИЕ. ЭЛЕКТРОННЫХ РЕСУРСОВ. Общие требования и правила составления [Текст]. – Введ. 01.06.2002. – Минск: Межгосударственный совет по стандартизации, метрологии и сертификации, 2007. – 27 с. – (Система стандартов по информации, библиотечному и издательскому делу).

6. ГОСТ 7.12-93 БИБЛИОГРАФИЧЕСКАЯ ЗАПИСЬ. СОКРАЩЕНИЕ СЛОВ НА РУССКОМ ЯЗЫКЕ. Общие требования и правила составления [Текст]. – Введ. 01.07.1995. – Минск: Межгосударственный совет по стандартизации, метрологии и сертификации, 2001. – 4 с. – (Научно-техническая

информация, библиотечное и издательское дело).

7. ГОСТ 7.0.5-2008. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая ссылка. Общие требования и правила составления [Текст]. Введ. 01.01.2009. – М: Стандартинформ, 2009. – 23 с. – (Система стандартов по информации, библиотечному и издательскому делу).

8. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения [Текст]. – Взамен ГОСТ 24.003-84, ГОСТ 22487-77; Введ. 01.01.1992. – М.: ГосСтандарт СССР, 1980. – 26 с.

9. Стандарт ISO/IEC 2382 Информационные технологии – Словарь – Первый Выпуск [Текст]. – Взамен ISO/IEC 2382-1:1993 – Женева, Швейцария, 2015. – 6 с.

10. Фот, А.П. Оформление библиографических ссылок на использованные источники в научных работах: методические рекомендации соискателям ученых степеней [Электронный документ] / А.П. Фот. – Оренбург: ГОУ «ОГУ», 2005. – 24 с. (<http://www.osu.ru/docs/official/nauka/bibllinks.doc>). Проверено 11.04.2010.

11. Олейник, П.П. Корпоративные информационные системы: для бакалавров и специалистов [Текст] / П.П. Олейник. – СПб: Издательский дом «Питер», 2012 – 174 с.

12. Радченко, М.Г. «1С:Предприятие 8.2». Практическое пособие разработчика. Примеры и типовые приемы [Текст] / М.Г. Радченко, Е.Ю. Хрусталева. – Москва: ООО «1С-Пабблишинг», 2013 г. – 964 с.

13. Китова, О.В. Корпоративные информационные системы управления [Текст] / О.В. Китова, Н.М. Абдикеев. – Москва: «ИНФРА-М», 2015 г. – 464 с.

14. Новикова, Г.М. Корпоративные информационные системы: Учеб. пособие [Текст] / Г.М. Новикова. – Москва: РУДН, 2008. – 94 с.

15. Бочаров, Е.П. Интегрированные корпоративные информационные системы [Текст] / Е.П. Бочаров, А.И. Колдина. – М.: Финансы и статистика, 2010.

– 350 с.

16. Рязанцева, Н.А. 1С: Предприятие. Комплексная конфигурация. Секреты работы [Текст] / Н.А. Рязанцева. – СПб: БХВ-Петербург, 2013 г. – 624 с.

17. Гудков, П.А. Методы сравнительного анализа: Учебное пособие [Текст] / П.А. Гудков. – Пенза: Издательство пензенского государственного университета, 2008 г. – 81 с.: ил.

18. Уварова, В.И. Использование метода анализа иерархий. Учебник [Текст] / В.И. Уварова, В.Г. Шуметов. – Изд-во Орел: Орел ГТУ, 2008 г. – 109 с.

19. Абакаров, А.Ш. Двухэтапная процедура отбора перспективных альтернатив на базе табличного метода и метода анализа иерархий [Текст] / А.Ш. Абакаров, Ю.А. Сушков – Изд-во СпбГУ, 2008 г. – 135 с.

20. Гладкий, А. В. 1С:Розница 8.2 с нуля. 50 уроков для начинающих [Текст] / А.В. Гладкий – Изд-во БХВ-Петербург, 2012 г. – 161с.

21. Селищев, Н. В. 1С:Розница. Оперативный и управленческий учет на компьютере [Текст] / Н.В. Селищев – Изд-во Рид Групп, 2011 г. – 368 с.

22. Гончаров, Д.И. Технологии интеграции 1С:Предприятия. Серия «1С:Профессиональная разработка» [Текст] / Д.И. Гончаров, Е.Ю. Хрусталева. – Изд-во Издательство «1С-Публишинг», 2011 г. – 358 с.

23. Мичасова, О.В. Имитационное моделирование экономических систем: Учебно-методическое пособие [Текст] / О.В. Мичасова. – Нижний Новгород: Нижегородский госуниверситет, 2014. – 186 с.

24. Ломазов, В.А. Когнитивная модель процесса принятия решения при выборе методов оценивания IT-проектов [Текст] / Ломазов В.А., Маторин С.И., Нехотина В.С. [Текст]. – 2015. – № 6-3. – С. 490-496; URL: <https://fundamental-research.ru/ru/article/view?id=38647> (дата обращения: 10.04.2017).

25. Радченко, М. Г. Профессиональная разработка в системе 1С: Предприятие 8". Издание 2 [Текст] / Радченко М. Г., Ажеронок В.А., Габец А.П., Гончаров Д.И., Козырев Д.В., Кухлевский Д.С., Островерх А.В., Хрусталева Е.Ю. [Текст]. – М.: ООО «1с-Публишинг», 2012 г. – 704 с.: ил.

26. Радченко, М. Г. Архитектура и работа с данными «1С: Предприятия

8.2». Серия «1С: Профессиональная разработка» [Текст] / М. Г. Радченко, Е. Ю. Хрусталева. – Москва: ООО «1С-Публишинг», 2011 г. – 268 с.: ил.

27. Маклаков, С.В. Моделирование бизнес-процессов с BРwin 4.0 [Текст] / С.В. Маклаков. – М.: ДИАЛОГМИФИ, 2011 г. – 24 с.

28. Репин, В.А. Процессный подход к управлению. Моделирование бизнес-процессов [Текст] / В.А. Репин, В.С. Елиферов. – Манн, Иванов и Фербер, 2013 г. – 17 с.

29. Маклаков, С. В. Моделирование бизнес-процессов с APFusion Process Modeler [Текст] / С.В. Маклаков. – Москва: Диалог-МИФИ, 2008 г. – 240 с.

30. Репин, В. В. Бизнес-процессы компании. Построение, анализ, регламентация [Текст] / В.В. Репин. – Москва: РИА «Стандарты и качество», 2008 г. – 240 с.: ил.

31. Бартенъев, О. 1С: Предприятие. Программирование для всех [Текст] / О. Бартенъев. – Москва: Диалог МИФИ, 2010 г. – 464 с.

32. Бояркин, В.Э. 1С: Предприятие 8. Конвертация данных: обмен данными между прикладными решениями [Текст] / В.Э. Бояркин, А.И. Филатов. – Москва: 1С: Публишинг, 2008 г. – 180 с.

33. Кашаев, С. М. 1С: Предприятие 8.3. Разработка прикладных решений [Текст] / С.М. Кашаев. – Москва: Вильямс, 2009 г. – 368 с.

34. Габец, А.П. Реализация прикладных задач в системе «1 С: Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталёва [Текст]. – Москва: ООО «1С-Публишинг», 2012 г. – 714 с.

35. Хрусталева, Е.Ю. Язык запросов "1С: Предприятия 8" [Текст] / Е.Ю. Хрусталева, М.Г. Радченко. – Москва: ООО «1С-Публишинг», 2013 г. – 358 с.

36. Кашаев, С. М. Программирование в 1С:Предприятие 8.3 [Текст] / С.М. Кашаев – Москва: Питер, 2014 г. – 304 с.

37. Бариленко, В. И. Комплексный анализ хозяйственной деятельности предприятия: учеб. пособие [Текст] / В.И. Бариленко. – Москва: ФОРУМ, 2012 г. – 463 с.

38. Громов, А.И. Управление бизнес-процессами: современные методы. монография [Текст] / А.И. Громов, А. Фляйшман, В. Шмидт. – Люберцы: Юрайт, 2016 г. – 367 с.
39. Долганова, О.И. Моделирование бизнес-процессов: Учебник и практикум для академического бакалавриата [Текст] / О.И. Долганова, Е.В. Виноградова, А.М. Лобанова. – Люберцы: Юрайт, 2016 г. - 289 с.
40. Толпегина, О.А. Комплексный экономический анализ хозяйственной деятельности [Текст] / О.А. Толпегина, Н.А. Толпегина – Москва: Юрайт, 2013 г. – 672 с.
41. Вдовин, В.М. Предметно-ориентированные экономические информационные системы: учеб.пособие [Электронный ресурс] / В.М. Вдовин, Л.Е. Суркова, А.А. Шурупов. – Москва: Дашков и К, 2013 г. – 385 с.
42. Цивенко, М. Ю. Методы и методика анализа имущественного положения предприятия и их характеристика [Электронный ресурс] / М.Ю. Цивенко. – 2017 г. – 105 с. – Режим доступа: <https://moluch.ru/archive/182/46768/> (дата обращения: 13.06.2018).
43. График спроса и предложения [Электронный ресурс] – Режим доступа: <http://exceltable.com/grafiki/spros-i-predlozhenie-v-excel>.
44. Базовые стандарты КИС: MRP, MRP II, ERP, ERP II и др. [Электронный ресурс]. – Режим доступа: http://studopedia.ru/7_188994_bazovie-standarti-kis-MRP-MRP-II-ERP-ERP-II-i-dr.html.
45. «1С:Предприятие» [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/1С:Предприятие>.
46. Официальный сайт компании «1С» [Электронный ресурс]. – Режим доступа: <http://v8.1c.ru/>.
47. Справочник «Внедренные решения». Список авторов решений [Электронный ресурс]. – Режим доступа: <http://1c.ru/rus/partners/solutions/partners.jsp?city=%C1%E5%EB%E3%EE%F0%E%E4>.
48. Преимущества и недостатки CRM-систем [Электронный ресурс]. –

Режим доступа: http://life-prog.ru/1_33645_preimushchestva-i-nedostatki-CRM-sistem.html.

49. CRM Системы [Электронный ресурс]. – Режим доступа: <http://bolsheprodaj.com/organizaciya-biznesa-i-prodazh/crm-sistemy.html>.

50. «Розничная торговля» [Электронный ресурс] – Режим доступа http://investments.academic.ru/1358/розничная_торговля.

51. Преимущества и недостатки ERP системы [Электронный ресурс]. – Режим доступа: http://www.kpms.ru/Automatization/ERP_system.htm.

52. Фирма «1С» [Электронный ресурс]. – Режим доступа: <http://1c.ru/>.

53. Фирма «Галактика» [Электронный ресурс]. – Режим доступа: <https://www.galaktika.ru/erp/upravlenie-logistikoj.html>.

54. Фирма «Парус» [Электронный ресурс]. – Режим доступа: <http://www.parus.com/solutions/middle/products/parus7/>.

55. Oracle [Электронный ресурс]. – Режим доступа: <https://oracle-patches.com/>.

56. SAP [Электронный ресурс]. – Режим доступа: <https://www.sap.com/index.html>.

57. Microsoft Dynamics 365 [Электронный ресурс]. – Режим доступа: <https://dynamics.microsoft.com/ru-ru/>.

58. Сравнение ERP-систем [Электронный ресурс]. – Режим доступа: http://ko.com.ua/erp_bitva_titanov_2017_119484.

59. Регламентные и фоновые задания в «1С:Предприятие» [Электронный ресурс]. – Режим доступа: <https://1s83.info/programmirovanie/reglamentnyie-i-fonovyie-zadaniya-v-1s.htm>.

60. GET и POST-запросы из «1С:Предприятие 8» [Электронный ресурс]. – Режим доступа: <http://ryakovlev.blogspot.com/2014/11/1-get-post-1-8.html>.

ПРИЛОЖЕНИЕ

Обработка «Juice Development» модуль формы «Клиент».

Перем мНомерКарты Экспорт;
Перем мДанныеКлиента Экспорт;
Перем ВвестиНовогоКлиента Экспорт;
Перем ВведенаКарта Экспорт;
Перем НовыйНомер;

Процедура ВидимостьДоступность()

Если ЗначениеЗаполнено(ВыдатьКарту) Тогда
 ЭлементыФормы.НовыйНомерКарты.Видимость = Истина;
 ЭлементыФормы.НадписьНоваяКарта.Видимость = Истина;

 ЭлементыФормы.НадписьНовыйКлиент.Заголовок = "ВЫДАЙТЕ КЛИЕНТУ НОВУЮ КАРТУ " +
СОКРЛП(ВыдатьКарту.ТипКарты);
Иначе
 ЭлементыФормы.НовыйНомерКарты.Видимость = Ложь;
 ЭлементыФормы.НадписьНоваяКарта.Видимость = Ложь;
КонецЕсли;

КонецПроцедуры

Функция ПоддерживаетсяВидГО(Вид) Экспорт

 Результат = Ложь;

 Если Вид = Перечисления.ВидыТорговогоОборудования.СканерШтрихКода Тогда
 Результат = Истина;
 КонецЕсли;

 Возврат Результат;

КонецФункции // ПоддерживаетсяВидГО()

Функция ПолучитьПараметрыНовогоКлиента(СтруктураДанныхОКлиенте)

 Возврат ЗаписанКлиент;

КонецФункции

Функция ДобавитьДКВСистему()

 СтруктураДанныхОКлиенте = Новый Структура;

 Если ВведенаКарта Тогда

 Если ЗначениеЗаполнено(ВыдатьКарту) Тогда
 СтруктураДанныхОКлиенте.Вставить("КодДК", СокрЛП(НовыйНомерКарты));

 Иначе
 СтруктураДанныхОКлиенте.Вставить("КодДК", СокрЛП(НомерКарты));
 КонецЕсли;

 Если ЗначениеЗаполнено(НовыйНомер) Тогда

 СтруктураДанныхОКлиенте.Вставить("НомерТелефона", НовыйНомер);
 КонецЕсли;

Иначе

 Если ЗначениеЗаполнено(НомерТелефона) Тогда
 СтруктураДанныхОКлиенте.Вставить("НомерТелефона", ЭлементыФормы.НомерТелефона.Значение);

КонецЕсли;
КонецЕсли;

Если ЗначениеЗаполнено(Фамилия) Тогда
 СтруктураДанныхОКлиенте.Вставить("Фамилия", СокрЛП(Фамилия));

КонецЕсли;
Если ЗначениеЗаполнено(Имя) Тогда
 СтруктураДанныхОКлиенте.Вставить("Имя", СокрЛП(Имя));
КонецЕсли;

Если ЗначениеЗаполнено(ДатаРождения) Тогда
 СтруктураДанныхОКлиенте.Вставить("ДатаРождения", ДатаРождения);
КонецЕсли;

Почта = СокрЛП(Почта);
 Если ЗначениеЗаполнено(Почта) Тогда
 СтруктураДанныхОКлиенте.Вставить("Почта",
Почта);
 КонецЕсли;

Если ЗначениеЗаполнено(Пол) Тогда
 СтруктураДанныхОКлиенте.Вставить("Пол",Пол);
КонецЕсли;

КлиентЗаписан = ПолучитьПараметрыНовогоКлиента(СтруктураДанныхОКлиенте);
Если КлиентЗаписан = Истина Тогда
 Если НЕ ЗначениеЗаполнено(ВыдатьКарту) Тогда
 Если ВведенаКарта Тогда
 ВладелецФормы.ОбъектВладелец.НомерКартыSailPlay = НомерКарты;
 Иначе
 ВладелецФормы.ОбъектВладелец.НомерТелефонаSailPlay = НомерТелефона;
 КонецЕсли;
 ВладелецФормы.ВладелецФормы.КартаПолученаПоНомеруТелефонаSailPlay = НЕ ВведенаКарта;
 КонецЕсли;
 КонецЕсли;
 Возврат КлиентЗаписан;
КонецФункции // ()

Процедура КнопкаВыполнитьНажатие1(Кнопка)

 ЧистыйНомер = SailPlay_Модуль.ВернутьЦифрыИзСтроки(ЭлементыФормы.НомерТелефона.Значение);

 Если ЗначениеЗаполнено(мДанныеКлиента) Тогда
 НомерТелефонаSailPlay = SailPlay_Модуль.ВернутьЦифрыИзСтроки(мДанныеКлиента.phone);
 НомерКартыSailPlay = мДанныеКлиента.origin_user_id;

 // запретим обнулять номер карты

 Если НЕ ЗначениеЗаполнено(НомерКарты) И ЗначениеЗаполнено(НомерКартыSailPlay) Тогда
 Предупреждение("Нельзя обнулять номер действующей карты!");
 Возврат;
 КонецЕсли;

 флСоздатьКарту = Ложь;

 Если СОКРЛП(НомерКарты) <> СокрЛП(НомерКартыSailPlay) Тогда // изменился номер карты
 Если ЗначениеЗаполнено(НомерКарты) И ЗначениеЗаполнено(НомерКартыSailPlay) Тогда
 отв = Вопрос("ВНИМАНИЕ! Изменился номер карты! Перенести данные на новую карту?",
РежимДиалогаВопрос.ДаНет);
 Если отв = КодВозвратаДиалога.Нет Тогда
 Возврат;
 КонецЕсли;
 КонецЕсли;

```

ДанныеКлиентаПоДК = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(НомерКарты,);
Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоДК)
    ИЛИ ДанныеКлиентаПоДК.status = "error" Тогда
        флКартаЗарегистрирована = Ложь; // карта не зарегистрирована в системе
        флСоздатьКарту = Истина;
Иначе
    флКартаЗарегистрирована = Истина;
КонецЕсли;

Если флСоздатьКарту Тогда
    /// создаем новую карту
    СтруктураДанныхОКлиенте = Новый Структура;
    СтруктураДанныхОКлиенте.Вставить("КодДК", СокрЛП(НомерКарты));
    КартаДобавлена = ПолучитьПараметрыНовогоКлиента(СтруктураДанныхОКлиенте);
    Если КартаДобавлена = Ложь Тогда
        Предупреждение("Не удалось зарегистрировать карту в системе!");
        Возврат;
    КонецЕсли;
КонецЕсли;

// если старый номер ДК заполнен - переносим данные на новую карты со старой
Если ЗначениеЗаполнено(НомерКартыSailPlay) Тогда
    СтруктураДанныхОКлиенте = Новый Структура;
    // при наличии заполненных одинаковых полей побеждает источник
    СтруктураДанныхОКлиенте.Вставить("НаКарту",НомерКартыSailPlay);
    СтруктураДанныхОКлиенте.Вставить("СКарты",НомерКарты);
    ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте,Истина);
    Если ДанныеВыгружены=Ложь Тогда
        Предупреждение("Не удалось перенести историю на новую карту!");
        Возврат;
    КонецЕсли;
    мДанные = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(НомерКарты);
    НомерТелефонаSailPlay = SailPlay_Модуль.ВернутьЦифрыИзСтроки(мДанныеКлиента.phone);
    НомерКартыSailPlay = мДанныеКлиента.origin_user_id;
КонецЕсли;

КонецЕсли;

// изменился номер телефона у клиента зарегистрированного по НТ
Если НЕ НомерТелефонаSailPlay = ЧистыйНомер И НЕ ЗначениеЗаполнено(НомерКартыSailPlay) Тогда
    ДанныеКлиентаПоНТ = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(ЧистыйНомер);
    Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ) // новый номер не зарегистрирован в СП
        ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
        СтруктураДанныхОКлиенте = Новый
Структура("НомерТелефона,НовыйНомер",НомерТелефонаSailPlay,ЧистыйНомер);
        ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlay(СтруктураДанныхОКлиенте,Истина);
        Если НЕ ДанныеВыгружены Тогда
            Предупреждение("Не удалось изменить номер телефона!");
            Возврат;
        КонецЕсли;
    КонецЕсли;
КонецЕсли;
КонецЕсли;

Если ЗначениеЗаполнено(НомерКарты) И ЗначениеЗаполнено(ЧистыйНомер) Тогда

    флКартаНеЗарегистрирована = Ложь;
    НТПривязанККарте = "";
    флНТНеЗарегистрирован = Ложь;
    ДКПривязанаКНТ = "";
    ДанныеКлиентаПоДК = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(НомерКарты,);

```



```

ДанныеКлиентаПоНТ = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(,ЧистыйНомер);

Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоДК)
    ИЛИ ДанныеКлиентаПоДК.status = "error" Тогда
    флКартаНеЗарегистрирована = Истина;           // карта не зарегистрирована в системе
Иначе
    phone = СокрЛП(ДанныеКлиентаПоДК.phone);
    ЧистыйНомерДК = SailPlay_Модуль.ВернутьЦифрыИзСтроки(phone);
    НТПривязанККарте = ЧистыйНомерДК;           // номер телефона привязанный к карте
КонецЕсли;

Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ)
    ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
    флНТНеЗарегистрирован = Истина;           // номер телефона не зарегистрирован
Иначе
    origin_user_id = ДанныеКлиентаПоНТ.origin_user_id;
    Если origin_user_id = "null" Тогда
        origin_user_id = "";
    КонецЕсли;
    ДКПривязанаКНТ = origin_user_id;           // карта привязанная к номеру телефона
КонецЕсли;

Если флКартаНеЗарегистрирована           // вводим новые регистрационные данные по новому
клиенту
    И флНТНеЗарегистрирован Тогда
        НовыйНомер = ЧистыйНомер;
    ИначеЕсли флКартаНеЗарегистрирована = Ложь           // привязываем к карте новый номер телефона
        И флНТНеЗарегистрирован И НЕ НТПривязанККарте = ЧистыйНомер Тогда
            НовыйНомер = ЧистыйНомер;
        ИначеЕсли флКартаНеЗарегистрирована = Ложь           // номер телефона уже привязан к данной карте -
просто обновляем данные клиента
            И флНТНеЗарегистрирован=Ложь И НТПривязанККарте = ЧистыйНомер Тогда
                ИначеЕсли (флКартаНеЗарегистрирована=Ложь И флНТНеЗарегистрирован=Ложь           // у карты номера
телефона карты не совпадает с текущим номером телефона
                    И НЕ ДКПривязанаКНТ = НомерКарты) Тогда
                    отв = Вопрос("Номер телефона привязан к другой карте!!! Привязать данный номер телефона к текущей
карте ?", РежимДиалогаВопрос.ДаНет);
                Если отв = КодВозвратаДиалога.Нет Тогда
                    НомерТелефона = "";
                    ЧистыйНомер = "";
                Иначе
                    СтруктураДанныхОКлиенте = Новый Структура;

                    // при наличии заполненных одинаковых полей побеждает источник
                    СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона",ЧистыйНомер);
                    СтруктураДанныхОКлиенте.Вставить("СКарты",НомерКарты);

                ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте,Истина);
                Если ДанныеВыгружены=Ложь Тогда
                    Предупреждение("Не удалось привязать телефон к карте!");
                    Возврат;
                КонецЕсли;
            КонецЕсли;
        ИначеЕсли (флКартаНеЗарегистрирована           // карта не зарегистрирована, НТ -
зарегистрирован и привязан к другой карте
            И флНТНеЗарегистрирован=Ложь И ЗначениеЗаполнено(ДКПривязанаКНТ)) Тогда
                отв = Вопрос("Номер телефона привязан к другой карте!!! Привязать данный номер телефона к текущей
карте ?", РежимДиалогаВопрос.ДаНет);
            Если отв = КодВозвратаДиалога.Нет Тогда
                НомерТелефона = "";
                ЧистыйНомер = "";
                КлиентЗаписан = ДобавитьДКВСистему();
            КонецЕсли;
        Иначе
            СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона",ЧистыйНомер);
            СтруктураДанныхОКлиенте.Вставить("СКарты",НомерКарты);
        КонецЕсли;
    КонецЕсли;

```

```

Иначе
    СтруктураДанныхОКлиенте = Новый Структура;
    СтруктураДанныхОКлиенте.Вставить("КодДК", СокрЛП(НомерКарты));
    КартаДобавлена = ПолучитьПараметрыНовогоКлиента(СтруктураДанныхОКлиенте);
    Если КартаДобавлена = Ложь Тогда
        Предупреждение("Не удалось зарегистрировать карту в системе!");
        Возврат;
    КонецЕсли;

    СтруктураДанныхОКлиенте = Новый Структура;
    // при наличии заполненных одинаковых полей побеждает источник
    СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона", ЧистыйНомер);
    СтруктураДанныхОКлиенте.Вставить("СКарты", НомерКарты);
    ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаVSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте, Истина);
    Если ДанныеВыгружены=Ложь Тогда
        Предупреждение("Не удалось привязать телефон к карте!");
        Возврат;
    КонецЕсли;
    //КлиентЗаписан = ДобавитьДКВСистему();
    КонецЕсли;
    ИначеЕсли (флКартаНеЗарегистрирована // карта не зарегистрирована, НТ -
зарегистрирован и не привязан к другой карте
    И флНТНеЗарегистрирован=Ложь И НЕ ЗначениеЗаполнено(ДКПривязанаКНТ)) Тогда
        НовыйНомер = ЧистыйНомер;
    КонецЕсли;

КонецЕсли;

КлиентЗаписан = ДобавитьДКВСистему();
Если КлиентЗаписан = Истина Тогда
    ВладелецФормы.НомерКарты = НомерКарты;
    ВладелецФормы.НомерТелефона = ЧистыйНомер;
    Закрыть();
    КонецЕсли;
КонецПроцедуры

Процедура КнопкаВыполнитьНажатие(Кнопка)

    ЧистыйНомер =
SailPlay_Модуль.ВернутьЦифрыИзСтроки(ЭлементыФормы.НомерТелефона.Значение);
    Если ЧистыйНомер = "7" Тогда
        ЧистыйНомер = "";
    КонецЕсли;

    ТекНомерКарты = "";
    ТекНомерТелефона = "";

    Если ЗначениеЗаполнено(мДанныеКлиента) Тогда
        ТекНомерТелефона = SailPlay_Модуль.ВернутьЦифрыИзСтроки(мДанныеКлиента.phone);
        ТекНомерКарты = СокрЛП(мДанныеКлиента.origin_user_id);
        Если НРег(ТекНомерКарты) = "null" Тогда
            ТекНомерКарты = "";
        КонецЕсли;
    КонецЕсли;

    Если ЗначениеЗаполнено(ВыдатьКарту) Тогда
        Если НЕ ЗначениеЗаполнено(НовыйНомерКарты) Тогда
            Предупреждение("Не заполнен код новой карты!!!");
            Возврат;
        КонецЕсли;
    КонецЕсли;

```

```

КлиентЗаписан = ложь;

Если ЗначениеЗаполнено(ВЫдатьКарту) Тогда
    Если НЕ ЗначениеЗаполнено(НовыйНомерКарты) Тогда
        Предупреждение("Не заполнен код новой карты!!!");
        Возврат;
    КонецЕсли;

флТелефонЗарегистрирован = Ложь;

Если НЕ флТелефонЗарегистрирован Тогда
    НовыйНомер = ЧистыйНомер;
КонецЕсли;

КартаДобавлена = ДобавитьДКВСистему();
Если КартаДобавлена = Ложь Тогда
    Предупреждение("Не удалось зарегистрировать карту в системе!");
    Возврат;
КонецЕсли;

Если флТелефонЗарегистрирован или флКартаЗарегистрирована Тогда
    СтруктураДанныхОКлиенте = Новый Структура;

    // при наличии заполненных одинаковых полей побеждает источник
    Если флКартаЗарегистрирована Тогда
        СтруктураДанныхОКлиенте.Вставить("НаКарту",НомерКарты);
    Иначе
        СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона",ЧистыйНомер);
    КонецЕсли;

    СтруктураДанныхОКлиенте.Вставить("СКарты",НовыйНомерКарты);
    ДанныеОбновлены =
SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте,Истина);
    Если ДанныеОбновлены=Ложь Тогда
        Предупреждение("Не удалось привязать перенести данные о бонусах на новую карту! Продолжайте
работать данные будут перенесены позже!");
        Возврат;
    КонецЕсли;
КонецЕсли;

ИначеЕсли ЗначениеЗаполнено(НомерКарты) Тогда
    Если ТекНомерКарты = НомерКарты Тогда
        Если НЕ ТекНомерТелефона = ЧистыйНомер Тогда
            Если ЧистыйНомер = "" Тогда
                Предупреждение("Не заполнен номер телефона!");
                Возврат;
            КонецЕсли;
        ИначеЕсли ЗначениеЗаполнено(НомерКарты) Тогда
            Предупреждение("Не заполнен номер карты!");
            Возврат;
        КонецЕсли;
    ИначеЕсли ЗначениеЗаполнено(НомерТелефона) Тогда
        Предупреждение("Не заполнен номер телефона!");
        Возврат;
    КонецЕсли;

ДанныеКлиентаПоНТ = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(,ЧистыйНомер);
ДКПривязанаКНТ = "";
Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ)
    ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
    флНТЗарегистрирован = Ложь; // номер телефона не зарегистрирован
Иначе
    флНТЗарегистрирован = Истина;
    origin_user_id = ДанныеКлиентаПоНТ.origin_user_id;
    Если origin_user_id = "null" Тогда
        origin_user_id = "";
    КонецЕсли;
    ДКПривязанаКНТ = origin_user_id; // карта привязанная к номеру телефона
КонецЕсли;

Если флНТЗарегистрирован Тогда

```

```

Если НЕ ДКПривязанаКНТ = ТекНомерКарты Тогда
    Если ЗначениеЗаполнено(ДКПривязанаКНТ) Тогда // номер телефона привязан к другой карте
        Если Вопрос("Данный номер телефона привязан к другой карте! Перенести данные на текущую
        карту?", РежимДиалогаВопрос.ДаНет) = КодВозвратаДиалога.Нет Тогда
            Возврат;
            КонецЕсли;
            КонецЕсли;
            // надо объединить данные с телефона на номер карты
            СтруктураДанныхОКлиенте = Новый Структура;
            СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона", ЧистыйНомер); // ??? номер
            телефона поменяется?
            СтруктураДанныхОКлиенте.Вставить("СКарты", НомерКарты);
            ДанныеВыгружены =
            SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте, Истина);
            Если НЕ ДанныеВыгружены Тогда
                Предупреждение("Не удалось записать изменения! Попробуйте снова");
                Возврат;
                КонецЕсли;
                НовыйНомер = ЧистыйНомер; // обновляем номер телефона
                КонецЕсли;
            Иначе
                // делаем update
                НовыйНомер = ЧистыйНомер; // обновляем номер телефона
                КонецЕсли;
            КонецЕсли;
Иначе
    /// изменился номер карты
    ДанныеКлиентаПоДК = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(НомерКарты,);
    Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоДК)
        ИЛИ ДанныеКлиентаПоДК.status = "error" Тогда
            флКартаЗарегистрирована = Ложь; // карта не зарегистрирована в системе
        Иначе
            флКартаЗарегистрирована = Истина;
        КонецЕсли;

    Если ЗначениеЗаполнено(ТекНомерКарты) Тогда
        Если Вопрос("Изменился номер карты! Перенести данные на новую карту?",
        РежимДиалогаВопрос.ДаНет) = КодВозвратаДиалога.Нет Тогда
            Возврат;
            КонецЕсли;
            КлиентЗаписан = Ложь;
            Если НЕ флКартаЗарегистрирована Тогда
                // надо создать карту
                КлиентЗаписан = ДобавитьДКВСистему();
                Если НЕ КлиентЗаписан Тогда
                    Предупреждение("Не удалось записать данные клиента! Попробуйте снова!");
                    Возврат;
                    КонецЕсли;
                КонецЕсли;
            КонецЕсли;

            // надо объединить данные двух карт
            СтруктураДанныхОКлиенте = Новый Структура;
            СтруктураДанныхОКлиенте.Вставить("НаКарту", ТекНомерКарты);
            СтруктураДанныхОКлиенте.Вставить("СКарты", НомерКарты);
            ДанныеВыгружены =
            SailPlay_Модуль.ОбновитьДанныеКлиентаBSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте, Истина);
            Если НЕ ДанныеВыгружены Тогда
                Предупреждение("Не удалось записать изменения! Попробуйте снова");
                Возврат;
                КонецЕсли;

            Если ЗначениеЗаполнено(ЧистыйНомер) Тогда
                ДанныеКлиентаПоДК =

```

```

SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(НомерКарты.);
Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоДК)
ИЛИ ДанныеКлиентаПоДК.status = "error" Тогда
Предупреждение("Не удалось записать изменения! Попробуйте снова");
Возврат;
Иначе
phone = СокрЛП(ДанныеКлиентаПоДК.phone);
НомерТелефонаИзДК = SailPlay_Модуль.ВернутьЦифрыИзСтроки(phone);
Если НЕ НомерТелефонаИзДК = ЧистыйНомер Тогда // изменился номер телефона
НовыйНомер = ЧистыйНомер; // обновляем номер телефона
ДанныеКлиентаПоНТ =
SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(,ЧистыйНомер);
Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ)
ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
флНТЗарегистрирован = Ложь; // номер телефона не зарегистрирован
Иначе
флНТЗарегистрирован = Истина;
КонецЕсли;

Если флНТЗарегистрирован Тогда
// надо объединить данные с телефона на номер карты
СтруктураДанныхОКлиенте = Новый Структура;
СтруктураДанныхОКлиенте.Вставить("НаНомерТелефона",ЧистыйНомер);
СтруктураДанныхОКлиенте.Вставить("СКарты",НомерКарты);
ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаВSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте,Истина);
Если НЕ ДанныеВыгружены Тогда
Предупреждение("Не удалось записать изменения! Попробуйте снова");
Возврат;
КонецЕсли;
НовыйНомер = ЧистыйНомер;
КонецЕсли;
КонецЕсли;
КонецЕсли;
Иначе
// клиент зарегистрирован по телефону

КлиентЗаписан = Ложь;

ДанныеКлиентаПоНТ = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(,ЧистыйНомер);
Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ)
ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
флНТЗарегистрирован = Ложь; // номер телефона не зарегистрирован
Иначе
флНТЗарегистрирован = Истина;
КонецЕсли;

Если НЕ флКартаЗарегистрирована Тогда
// надо создать карту
Если НЕ флНТЗарегистрирован Тогда
НовыйНомер = ЧистыйНомер;
КонецЕсли;
КлиентЗаписан = ДобавитьДКВСистему();
Если НЕ КлиентЗаписан Тогда
Предупреждение("Не удалось записать данные клиента! Попробуйте снова!");
Возврат;
КонецЕсли;
КонецЕсли;

Если флНТЗарегистрирован Тогда
Если НЕ ДанныеКлиентаПоНТ.origin_user_id = НомерКарты Тогда
// надо объединить данные с телефона на номер карты

```

```

        СтруктураДанныхОКлиенте = Новый Структура;
        СтруктураДанныхОКлиенте.Вставить("НомерТелефона", ЧистыйНомер);
        СтруктураДанныхОКлиенте.Вставить("СКарты", НомерКарты);
        ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаВSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте, Истина);
        Если НЕ ДанныеВыгружены Тогда
            Предупреждение("Не удалось записать изменения! Попробуйте снова");
            Возврат;
        КонецЕсли;
        НовыйНомер = ЧистыйНомер; // обновляем номер телефона
        КонецЕсли;
    КонецЕсли;
КонецЕсли;
КонецЕсли;

Иначе

    /// если не заполнены номера карт то работаем только с телефонами
    Если ЗначениеЗаполнено(ТекНомерКарты) Тогда
        Предупреждение("Нельзя обнулять номер карты!");
        Возврат;
    КонецЕсли;

    Если ЧистыйНомер = ТекНомерТелефона Тогда // номер телефона не изменился
    Иначе
        Если НЕ ЗначениеЗаполнено(ЧистыйНомер) Тогда
            Предупреждение("Заполните номер телефона!");
            Возврат;
        КонецЕсли;

        КлиентЗаписан = Ложь;

        ДанныеКлиентаПоНТ = SailPlay_Модуль.ПолучитьИнформациюОКлиентеИзSailPlay(ЧистыйНомер);
        Если НЕ ЗначениеЗаполнено(ДанныеКлиентаПоНТ)
            ИЛИ ДанныеКлиентаПоНТ.status = "error" Тогда
                флНТЗарегистрирован = Ложь; // номер телефона не зарегистрирован
            Иначе
                флНТЗарегистрирован = Истина;
            КонецЕсли;

            Если ЗначениеЗаполнено(ТекНомерТелефона) И флНТЗарегистрирован Тогда
                Если Вопрос("Перенести данные на новый телефон клиента?", РежимДиалогаВопрос.ДаНет) =
КодВозвратаДиалога.Нет Тогда
                    Возврат;
                КонецЕсли;
            КонецЕсли;

            Если НЕ флНТЗарегистрирован Тогда
                // надо создать карту
                КлиентЗаписан = ДобавитьДКВСистему();
                Если НЕ КлиентЗаписан Тогда
                    Предупреждение("Не удалось записать данные клиента! Попробуйте снова!");
                    Возврат;
                КонецЕсли;
            КонецЕсли;

            Если ЗначениеЗаполнено(ТекНомерТелефона) Тогда
                // надо объединить данные с телефона на номер карты
                СтруктураДанныхОКлиенте = Новый Структура;
                СтруктураДанныхОКлиенте.Вставить("НомерТелефона", ТекНомерТелефона);
                СтруктураДанныхОКлиенте.Вставить("СКарты", ЧистыйНомер);
                ДанныеВыгружены =
SailPlay_Модуль.ОбновитьДанныеКлиентаВSailPlayСПереносомБонусов(СтруктураДанныхОКлиенте, Истина);

```

```

    Если НЕ ДанныеВыгружены Тогда
        Предупреждение("Не удалось записать изменения! Попробуйте снова");
        Возврат;
    КонечЕсли;
    НовыйНомер = ЧистыйНомер;
    КонечЕсли;
    КонечЕсли;
    КонечЕсли;

Если НЕ КлиентЗаписан Тогда
    КлиентЗаписан = ДобавитьДКВСистему();
    КонечЕсли;
Если КлиентЗаписан = Истина Тогда
    Если НЕ ЗначениеЗаполнено(ВыдатьКарту) Тогда
        ВладелецФормы.НомерКарты = НомерКарты;
        длНомер = СтрДлина(ЧистыйНомер);
        Если длНомер = 11 Тогда
            Если НЕ Лев(Прав(ЧистыйНомер,длНомер-1),1) = "7" Тогда
                ЧистыйНомер = Прав(ЧистыйНомер,длНомер-1);
            КонечЕсли;
        КонечЕсли;
        ВладелецФормы.НомерТелефона = ЧистыйНомер;
    КонечЕсли;
        Закрывать(Истина);
Иначе
    Предупреждение("Не удалось записать данные!");
    КонечЕсли;
КонечПроцедуры

Процедура ПередОткрытием(Отказ, СтандартнаяОбработка)
    РезультатПодключенияТО = ПолучитьСерверТО().ПодключитьКлиента(ЭтаФорма);
    Если ЗначениеЗаполнено(РезультатПодключенияТО) Тогда
        ТекстОшибки = "Ошибка подключения сканера. Сканирование номера карты недоступно. Ввод
        карты возможен через F7";
        Предупреждение(ТекстОшибки);
    КонечЕсли;

    НомерКарты          = мНомерКарты;
    ДанныеКлиента      = мДанныеКлиента;

    ЭлементыФормы.СтатусКарты.СписокВыбора.Добавить(Перечисления.СтатусКартыSailPlay.Базовая);
    ЭлементыФормы.СтатусКарты.СписокВыбора.Добавить(Перечисления.СтатусКартыSailPlay.Улучшенный);
    ЭлементыФормы.СтатусКарты.СписокВыбора.Добавить(Перечисления.СтатусКартыSailPlay.Премиум);

Если ВвестиНовогоКлиента = Ложь И ЗначениеЗаполнено(ДанныеКлиента) Тогда
    ЭлементыФормы.НадписьНовыйКлиент.Заголовок = "Редактирование данных клиента";
    Если ЗначениеЗаполнено(ДанныеКлиента.phone) И ДанныеКлиента.phone <> "null" Тогда
        НомерТелефонаSailPlay = ДанныеКлиента.phone;
        Если СтрДлина(НомерТелефонаSailPlay) = 11 Тогда
            НомерТелефона = Сред(НомерТелефонаSailPlay,2);
        Иначе
            НомерТелефона = НомерТелефонаSailPlay;
        КонечЕсли;
        НомерТелефона = ЭлементыФормы.НомерТелефона.Значение;
    КонечЕсли;
    Если ЗначениеЗаполнено(ДанныеКлиента.first_name) И ДанныеКлиента.first_name <> "null" Тогда
        Имя = ДанныеКлиента.first_name;
    КонечЕсли;
    Если ЗначениеЗаполнено(ДанныеКлиента.last_name) И ДанныеКлиента.last_name <> "null" Тогда
        Фамилия = ДанныеКлиента.last_name;
    КонечЕсли;

    Если ЗначениеЗаполнено(ДанныеКлиента.sex) И ДанныеКлиента.sex <> "null" Тогда

```

```
ПолКод = СОКРЛП(ДанныеКлиента.sex);
Если ПолКод = "1" Тогда
    Пол = Перечисления.ПолФизическихЛиц.Мужской;
ИначеЕсли ПолКод = "2" Тогда
    Пол = Перечисления.ПолФизическихЛиц.Женский;
КонецЕсли;
КонецЕсли;
Если ЗначениеЗаполнено(ДанныеКлиента.birth_date) И ДанныеКлиента.birth_date <> "null" Тогда
    ДатаРождения = Дата(СтрЗаменить(ДанныеКлиента.birth_date, "-", ""));
КонецЕсли;
СтатусКарты = ПолучитьСтатусКарты(НомерКарты,НомерТелефона);
КонецЕсли;
КонецПроцедуры
```

Процедура ПриОткрытии()

```
Если ЗначениеЗаполнено(НомерТелефона) Тогда
    ЧистыйНомер = SaiPlay_Модуль.ВернутьЦифрыИзСтроки(НомерТелефона);
    длНомер = СтрДлина(ЧистыйНомер);
    Если длНомер = 11 Тогда
        Если НЕ Лев(Прав(ЧистыйНомер,длНомер-1),1) = "7" Тогда
            ЧистыйНомер = Прав(ЧистыйНомер,длНомер-1);
        КонецЕсли;
    КонецЕсли;
    НомерТелефона = ЧистыйНомер;
    НомерТелефона = ЭлементыФормы.НомерТелефона.Значение;
КонецЕсли;
ВидимостьДоступность();
КонецПроцедуры
```

Процедура ПриИзмененииДК()

```
Если НЕ ЗначениеЗаполнено(НомерКарты) Тогда
    Возврат;
КонецЕсли;
КонецПроцедуры
```

Процедура СШКОбработатьШтрихкод(Штрихкод) Экспорт

```
Если ЗначениеЗаполнено(Штрихкод) Тогда
    Если ЗначениеЗаполнено(ВыдатьКарту) Тогда
        НовыйНомерКарты = СокрЛП(Штрихкод);
    Иначе
        НомерКарты = СокрЛП(Штрихкод);
    КонецЕсли;
    ВведенаКарта = Истина;
КонецЕсли;
КонецПроцедуры
```

Процедура ВнешнееСобытие(Источник, Событие, Данные)

```
Если Не ВводДоступен() Тогда
    Возврат;
КонецЕсли;
ПолучитьСерверТО().ОбработатьВнешнееСобытие(Источник, Событие, Данные, ЭтаФорма);
КонецПроцедуры
```

Процедура НомерТелефонаПриИзменении(Элемент)

```
НомерТелефона = ЭлементыФормы.НомерТелефона.Значение;
КонецПроцедуры
```

Процедура КоманднаяПанель1ВвестиШК(Кнопка)

```
Результат = РаботаСТорговымОборудованием.ВвестиШтрихкод();

Если ЗначениеЗаполнено(Результат) Тогда
    Если ЗначениеЗаполнено(ВыдатьКарту) Тогда
        НовыйНомерКарты = СокрЛП(Результат);
    КонецЕсли;
КонецЕсли;
```



```

Иначе
    НомерКарты = СокрЛП(Результат);
КонецЕсли;
ВведенаКарта = Истина;
КонецЕсли;

КонецПроцедуры

Процедура НовыйНомерКартыПриИзменении(Элемент)
    ВведенаКарта = Истина;
КонецПроцедуры

Процедура ПриЗакрытии()
    ПолучитьСерверТОО().ОтключитьКлиента(ЭтаФорма);
КонецПроцедуры

Общий модуль Отложенные транзакции

Процедура ВыгрузитьОтложенныеТранзакции(Ссылка = Неопределено, ЧислоПопыток) Экспорт

    Если Ссылка = Неопределено Тогда //тоесть когда плановая выгрузка отложенных транзакций, пробуем
    восстановить связь
        Если ПараметрыСеанса.ОткладыватьТранзакцииSailPlay Тогда
            ПараметрыСеанса.ОткладыватьТранзакцииSailPlay = Ложь; //каждые N минут
            пробуем наладить связь с SailPlay
            КонецЕсли;
        КонецЕсли;

        Запрос = Новый Запрос;
        Запрос.Текст = "ВЫБРАТЬ
            |         ОтложенныеТранзакцииSailPlay.Документ,
            |         ОтложенныеТранзакцииSailPlay.ВидОперации КАК ВидОперации
            |ИЗ
            |         РегистрСведений.ОтложенныеТранзакцииSailPlay КАК
ОтложенныеТранзакцииSailPlay
            |ГДЕ
            |         ОтложенныеТранзакцииSailPlay.ТранзакцияЗавершена = ЛОЖЬ
            |         И &УсловиеПоДокументу
            |         И ГОД(ОтложенныеТранзакцииSailPlay.Период) > 2017
            |         И ОтложенныеТранзакцииSailPlay.ЧислоПопыток МЕЖДУ &Начало И
&ЧислоПопыток
            |
            |УПОРЯДОЧИТЬ ПО
            |         ОтложенныеТранзакцииSailPlay.ВидОперации.Порядок,
            |         ОтложенныеТранзакцииSailPlay.Документ.Дата";

        Если Ссылка = Неопределено Тогда //накладываем фильтр
            Запрос.Текст = СтрЗаменить(Запрос.Текст, "&УсловиеПоДокументу", "Истина");
        Иначе
            Запрос.Текст =
СтрЗаменить(Запрос.Текст, "&УсловиеПоДокументу", "ОтложенныеТранзакцииSailPlay.Документ =
&Документ");
            Запрос.УстановитьПараметр("Документ", Ссылка);
        КонецЕсли;

        Запрос.УстановитьПараметр("ЧислоПопыток", ЧислоПопыток);
        Если ЧислоПопыток=5 тогда
            Запрос.УстановитьПараметр("Начало", 1);
        ИначеЕсли ЧислоПопыток=20 тогда
            Запрос.УстановитьПараметр("Начало", 6);
        Иначе
            Запрос.Текст = СтрЗаменить(Запрос.Текст, "МЕЖДУ &Начало И
&ЧислоПопыток", ">=&ЧислоПопыток");

```

```

        КонечЕсли;

    РезультатЗапроса = Запрос.Выполнить();

    Если РезультатЗапроса.Пустой() = Ложь Тогда

        Выборка = РезультатЗапроса.Выбрать();
        Пока Выборка.Следующий() Цикл

            Выгружен = Ложь;

            Если Выборка.ВидОперации = Перечисления.ВидыОперацииSailPlay.Продажа Тогда
                Выгружен =
                SailPlay_Модуль.ВыгруженаПродажаBSailPlayПоЧеку(Выборка.Документ,,Истина);
                ИначеЕсли Выборка.ВидОперации = Перечисления.ВидыОперацииSailPlay.Возврат
                Тогда
                    Выгружен =
                    SailPlay_Модуль.НачислитьСписатьБонусыBSailPlayПоВозврату(Выборка.Документ,Истина);
                    КонечЕсли;

                    Если Выгружен = Истина Тогда

                        НаборЗаписей =
                        РегистрыСведений.ОтложенныеТранзакцииSailPlay.СоздатьНаборЗаписей();
                        НаборЗаписей.Отбор.Документ.Установить(Выборка.Документ);
                        НаборЗаписей.Отбор.ВидОперации.Установить(Выборка.ВидОперации);
                        НаборЗаписей.Прочитать();

                        Для Каждого Запись Из НаборЗаписей Цикл
                            Запись.ТранзакцияЗавершена = Истина;
                            КонечЦикла;

                            НаборЗаписей.Записать();

                        Иначе
                            НаборЗаписей =
                            РегистрыСведений.ОтложенныеТранзакцииSailPlay.СоздатьНаборЗаписей();
                            НаборЗаписей.Отбор.Документ.Установить(Выборка.Документ);
                            НаборЗаписей.Отбор.ВидОперации.Установить(Выборка.ВидОперации);
                            НаборЗаписей.Прочитать();

                            Для Каждого Запись Из НаборЗаписей Цикл
                                Если ЗначениеЗаполнено(Запись.ЧислоПопыток) тогда
                                    Запись.ЧислоПопыток = Запись.ЧислоПопыток+1;
                                Иначе
                                    Запись.ЧислоПопыток = 1;
                                    КонечЕсли;
                                КонечЦикла;

                                НаборЗаписей.Записать();

                            КонечЕсли;

                        КонечЦикла;

        КонечЕсли;

    КонечПроцедуры

//отложенные транзакции

Процедура МеньшеПятиПопыток() Экспорт
    ВыгрузитьОтложенныеТранзакции(,5);

```

КонецПроцедуры

Процедура МеньшеДвадцатиПопыток() Экспорт
ВыгрузитьОтложенныеТранзакции(,20);
КонецПроцедуры

Процедура БольшеДвадцатиПопыток() Экспорт
ВыгрузитьОтложенныеТранзакции(,21);
КонецПроцедуры