

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

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Прикладная информатика в социальной сфере

(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему «**Разработка элементов CRM-системы для североамериканской телекоммуникационной компании**»

Студент

А.Ю. Кузьмин

(И.О. Фамилия)

(личная подпись)

Руководитель

Э.В. Егорова

(И.О. Фамилия)

(личная подпись)

Консультанты

И.Ю. Усатова

(И.О. Фамилия)

(личная подпись)

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

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

(личная подпись)

« _____ » _____ 20__ г.

Тольятти 2018

Аннотация

Тема выпускной квалификационной работы: «Разработка элементов CRM-системы для североамериканской телекоммуникационной компании».

ЭЛЕМЕНТЫ CRM-СИСТЕМЫ, АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ, ПРОЕКТИРОВАНИЕ, АВТОМАТИЗАЦИЯ.

Целью ВКР является разработка CRM-системы североамериканской телекоммуникационной компании для организации управления клиентскими отношениями.

Структура работы представлена введением, тремя главами, заключением, списком литературы и приложением.

Во введении описывается актуальность темы, формируется цель и задачи, которые необходимо выполнить.

В первой главе выполнен анализ деятельности компании, рассматриваются бизнес-процессы деятельности менеджера по продажам, ставится задача на автоматизацию.

Во второй главе описывается проектирование CRM-системы для североамериканской телекоммуникационной компании с помощью структурного и объектно-ориентированного подходов.

В третьей главе представлена технология разработки и принципы работы CRM-системы для североамериканской телекоммуникационной компании, приведено обоснование целесообразности использования CRM-системы для предприятия.

В заключении представлены результаты и выводы о выполненной работе.

Результатом выпускной квалификационной работы будет элемент CRM-системы для североамериканской телекоммуникационной компании. В приложении представлены фрагменты кода.

В работе представлено 6 таблиц 27 рисунков, список использованной литературы содержит 22 источника. Общий объем выпускной квалификационной работы составляет 57 страниц.

Abstract

The theme of the graduation work is «CRM-system for the North American Telecom Company».

The aim is the development of a CRM-system for management customer relations.

Structurally the work includes an introduction, three chapters, a conclusion and a list of references and an application.

In the first chapter, the analysis of sales manager activities of the North American Telecommunications Company is made, business processes are reviewed, and automation is put on the task.

The second chapter describes the design of the CRM-system with the help of structural and object-oriented approaches.

The third chapter presents the development technology and principles of the CRM-system, the rationale for using the CRM-system for the enterprise.

The conclusion presents the results and conclusions on the work performed.

The result of the work is a part of CRM-system, which automates customer relations by providing an online record and track the status of fulfilling its request.

The application contains snippets of code with detailed comments.

The work presents 6 tables, 27 figures, the list of references contains 22 sources. The total amount of the graduation work is 57 pages.

Оглавление

Введение.....	6
Глава 1 Функциональное моделирование CRM-системы для североамериканской телекоммуникационной компании.....	9
1.1 Техничко-экономическая характеристика компании.....	9
1.2 Краткая характеристика процесса управления заказами компании.....	10
1.3 Концептуальное моделирование бизнес-процесса.....	12
1.3.1 Выбор технологии концептуального моделирования автоматизированной информационной системы управления заказами североамериканской телекоммуникационной компании.....	12
1.3.2 Разработка и анализ модели бизнес-процесса управления заказами североамериканской телекоммуникационной компании «КАК ЕСТЬ»	14
1.3.3 Обоснование необходимости автоматизированного варианта решения и формирование требований к новой технологии.....	17
1.4 Анализ существующих разработок на предмет соответствия сформулированным требованиям	19
1.5 Постановка задачи на разработку проекта создания/внедрения автоматизированной информационной системы.....	22
1.6 Разработка модели <u>бизнес-процесса</u> «КАК <u>ДОЛЖНО</u> БЫТЬ».....	23
Выводы по главе 1	26
Глава 2 Логическое проектирование CRM-системы для североамериканской телекоммуникационной компании	27
2.1 Выбор технологий логического моделирования автоматизированной информационной системы управления заказами североамериканской телекоммуникационной компании	27
2.2 Логическая модель информационной системы и её описание	28
2.3 Разработка концептуальной и логической моделей данных информационной системы компании	31

2.4 Требования к аппаратно-программному обеспечению информационной системы	33
Выводы по главе 2	34
Глава 3 Практическая реализация CRM-системы	35
3.1 Выбор технологии разработки программного обеспечения.....	35
3.2 Описание основного принципа работы CRM-системы для североамериканской телекоммуникационной компании.....	36
3.3 Тестирование CRM-системы для североамериканской телекоммуникационной компании	39
3.4 Обоснование эффективности использования CRM-системы для североамериканской телекоммуникационной компании.....	41
Вывод по третьей главе.....	44
Заключение.....	46
Список используемой литературы	48
Приложение А	51

Введение

В настоящее время конкуренция в предоставлении услуг очень высокая, в связи с этим чтобы компания была высококорентабельной требуются особые отношения с клиентами.

На сегодня известно, что оптимизации производства и повышения качества продукции уже недостаточно, особенно в сфере услуг, так как отношения с клиентами играет такую же большую роль, как и качество продукции. Из этого следует, что для развития компании крайне важна продуктивная работа с клиентами, чтобы удержать и собрать новых клиентов.

Одна из главных ролей в компаниях отведена менеджерам по продажам. Именно они осуществляют взаимодействие с клиентами. Что бы упростить работу менеджеров по продажам существует CRM система. CRM-система предназначена для привлечения клиентов. Использование CRM-систем позволяет эффективно управлять взаимоотношениями с клиентами. CRM - это подход к self-care управлению, модель, которая позволяет клиенту самостоятельно управлять бизнес-процессами компании. Если клиентская база компании достаточно большая, то без программного обеспечения и серверов управлять отношениями с клиентами очень сложно. В связи с этим, целесообразно внедрение CRM-систем для автоматизации некоторых бизнес-процессов компании.

Актуальность выпускной квалификационной работы (ВКР) обусловлена необходимостью автоматизации управления взаимоотношениями с клиентами по заказу организации.

Объектом исследования является бизнес-процесс оформления заказов на установку телекоммуникационных услуг в североамериканской телекоммуникационной компании.

Предметом исследования является автоматизация бизнес-процесса управления заказами клиентов на установку телекоммуникационных услуг в североамериканской телекоммуникационной компании.

Целью ВКР является разработка автоматизированной информационной системы управления заказами клиентов с элементами CRM-системы для североамериканской телекоммуникационной компании.

Для достижения поставленной цели, необходимо решить следующие задачи:

- изучить предметную область;
- разработать концептуальную модель CRM-Системы;
- изучить и проанализировать аналоги CRM-Систем;
- разработать логическую модель данных CRM-Системы;
- реализовать CRM-Систему и базу данных системы;
- провести тестирование CRM-системы;
- обосновать эффективность использования системы.

При написании работы использовались методы исследования: анализ, системный подход, методы моделирования бизнес-процессов при помощи Case-средств. Так же в работе использовались учебные пособия, стандарты и материалы по моделированию и проектированию программных средств.

В данной выпускной квалификационной работе рассматриваются вопросы по разработке и реализации CRM системы североамериканской телекоммуникационной компании.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка использованных источников и приложения.

В первой главе описывается деятельность компании. Приводится сравнение аналогов CRM-систем, основываясь на полученных сведениях были сформулированы требования к разрабатываемой CRM-системе.

Во второй главе описывается логическое проектирование информационной системы. В ней происходит выбор технологии логического моделирования, а также проектирование БД, происходит построение логической модели. Описывает все выявленные требования к аппаратно-программному обеспечению автоматизированной информационной системы.

В третьей главе был проведён анализ основных средств реализации CRM системы для североамериканской телекоммуникационной компании. На основе результатов анализа основных средств реализации, были выбраны средства с помощью которых осуществлялась реализация CRM-системы. Также были описаны ход реализации и основные принципы работы CRM-системы. После реализации, демонстрируется тестирование методом «Чёрного ящика» и его результаты. После тестирования CRM-системы, доказывается пригодность системы для использования на предприятии.

В заключении приводятся результаты и выводы о проделанной работе.

Итогом выпускной квалификационной работы является разработанная CRM-система для североамериканской телекоммуникационной компании, предназначенная для автоматизации и управлением взаимоотношениями с клиентами.

Данная работа выполнялась по заказу компании. Результат работы планируется к внедрению основной системы компании.

Глава 1 Функциональное моделирование CRM-системы для североамериканской телекоммуникационной компании

1.1 Техничко-экономическая характеристика компании

Штаб-квартира располагается на территории северной части США. Также множество офисов располагаются по всей территории США, СНІ уже более 35 лет работают над предоставлением коммуникационных и технологических услуг для некоторых из самых отдаленных территорий, которые находятся в самых тяжелых условиях в Северной Америке. СНІ является первопроходцем в своей области, предоставляя телемедицину и онлайн-образовательные возможности сообществам по всему штату и продолжая усилия по глобальному подключению, предоставляя при этом услуги на потребительском и деловом рынках. Вступление СНІ в число провайдеров, предоставляющих оптоволокно в штате, а также инновационное партнерство с Apple - относятся к числу бесчисленных способов, которыми компания пытается улучшить общение и качество жизни для местных жителей.

В компании вопросами оформления и приема заказов занимается менеджер по работе с клиентами по телефону.

У компании есть свой сайт. На рисунке 1.1 представлен скриншот.

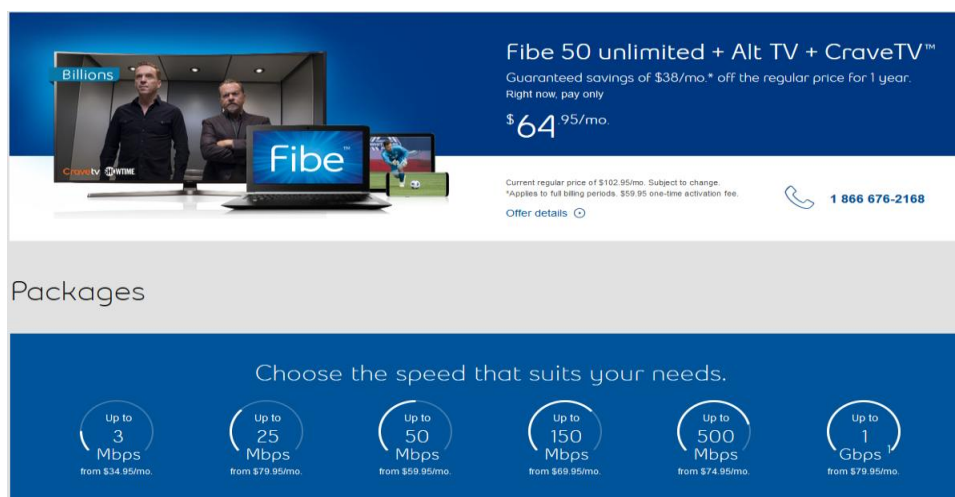


Рисунок 1.1 - Скриншот сайта компании

СНІ известна своими новаторскими усилиями в области телефонных и других телекоммуникационных коммерческих услуг, являясь первыми, кто

предложил комплекс продуктов и выдающиеся услуги для клиентов в Северной Америке. Чтобы поддерживать импульс на будущее развитие, СНИ работает в партнерстве с ведущими поставщиками технологий в Северной Америке.

1.2 Краткая характеристика процесса управления заказами компании

Бизнес-процесс управления заказами, играющий одну из ключевых ролей в клиенто-ориентированном сервисе, является обеспечивающим для основного бизнес-процесса оказания услуг по предоставлению телекоммуникационных услуг и состоит из следующих этапов:

- 1) первоначальный контакт с клиентом по телефону или непосредственно в офисе;
- 2) определение объема и сроков выполнения заявленных работ;
- 3) определение предварительной стоимости услуги;
- 4) согласование дат;
- 5) формирование предварительного заказа-наряда.

Модель процесса «Управление заказами» представлена на рисунке 1.2. На контекстной модели отображен бизнес-процесс основной деятельности менеджера компании.

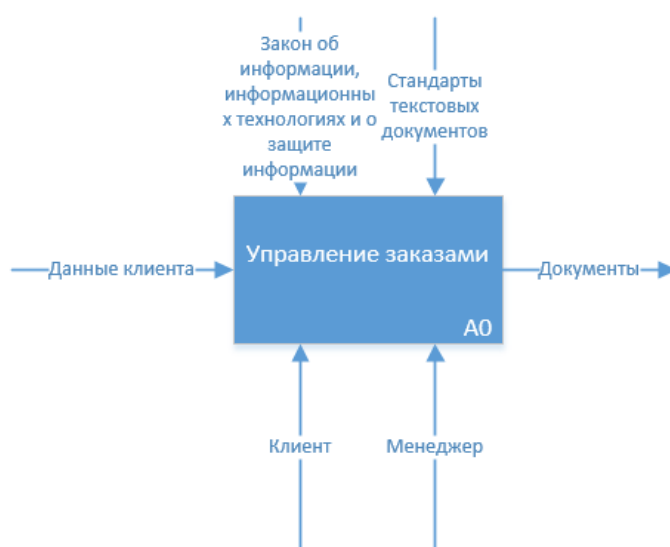


Рисунок 1.2 – Контекстная модель

На вход процесса «Управление заказами» поступают данные клиента и его заявка. На выходе из процесса А0 клиент получает нужные документы. Чтобы лучше понять структуру главного процесса необходимо разбить процесс на подпроцессы. Рассмотрим более подробно процессы декомпозиции:

- 1) процесс «Добавление нового клиента и просмотр существующих» подразумевает, что клиент подает заявку, а менеджер компании вносит полученные данные;
- 2) процесс «Добавление новой заявки» подразумевает, что менеджер организации вносит данные о новой заявке, делает расчет стоимости услуг и подсчитывает предоплату;
- 3) процесс «Формирование предварительного заказ-наряда» подразумевает выбор шаблона документа и заполнение его данными менеджером, далее обе стороны подписывают документ, и менеджер компании выдает копию документа клиенту. Этот процесс является заключительным.

Модель декомпозиции представлена на рисунке 1.3.

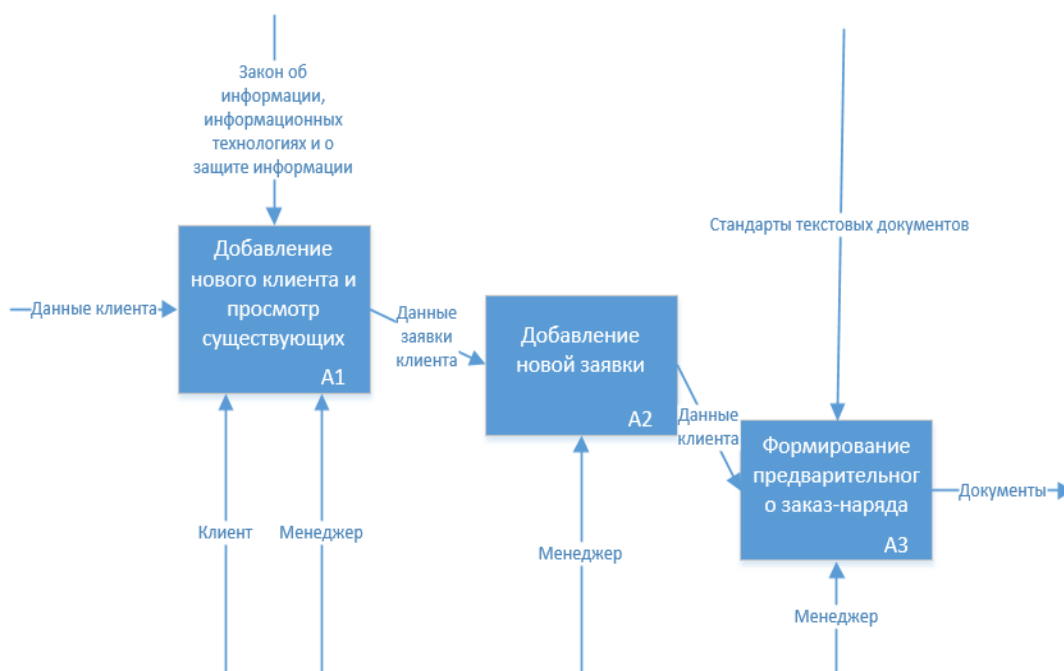


Рисунок 1.3 – Декомпозиция контекстной модели

Бизнес-процесс не автоматизирован. Далее перейдем к концептуальному моделированию бизнес-процесса.

1.3 Концептуальное моделирование бизнес-процесса

1.3.1 Выбор технологии концептуального моделирования автоматизированной информационной системы управления заказами североамериканской телекоммуникационной компании

Концептуальная модель является отраслевой моделью и разрабатывается для компании системным интегратором на основе методологии реинжиниринга и управления бизнес-процессами телекоммуникационной компании. Построение и оптимизация компании в методологии реинжиниринга осуществляется в 5 этапов:

- разработка модели AS-IS («КАК ЕСТЬ»);
- анализ модели «КАК ЕСТЬ»;
- разработка модели TO-BE («КАК ДОЛЖНО БЫТЬ»);
- разработка плана перехода из состояния («КАК ЕСТЬ» в состояние «КАК ДОЛЖНО БЫТЬ»);
- внедрение изменений и построение организации «КАК ДОЛЖНО БЫТЬ».

На сегодня для обеспечения высокой эффективности информационных систем рекомендуется использование различных подходов к их проектированию и моделированию. Для успешного выполнения процессного анализа используются методы и инструментальные средства, основанные на структурном подходе, основанные на принципе функциональной декомпозиции – выделении элементов предметной области, их взаимосвязей и семантических отношений. Выберем технологию концептуального моделирования, проведя сравнительный анализ трех наиболее известных нотаций концептуального моделирования информационной системы: BPMN, ARIS, UML.

BPMN (Business Process Model and Notation, нотация и модель бизнес-процессов) - система условных обозначений (нотация) для моделирования

бизнес-процессов. Разработана Business Process Management Initiative (BPMI.org) и поддерживается Object, после слияния обеих организаций в 2005 году.

UML (Unified Modeling Language) - унифицированный язык моделирования, который представляет собой графическую нотацию, предназначенную для моделирования и описания всех процессов, протекающих при разработке проекта.

ARIS (Architecture of Integrated Information Systems) - методология, а также комплекс средств, который формализует информацию, анализирует и оптимизирует деятельность предприятия и представляет ее в виде графических моделей.

Результаты анализа представим в таблице 1.1.

Таблица 1.1 – Сравнительный анализ методологий

Критерий сравнения	Нотация BPMN	Нотация ARIS	Нотация UML
Легкость изучения	Легко	Очень сложно	Сложно
Система хранения данных модели	Модели хранятся в файлах	Объектная база данных	Модели хранятся в файлах
Возможность декомпозиции	Неограниченная	Неограниченная	Неограниченная
Подход к проектированию	Функциональный	Процессный	Объектно-ориентированный

Из таблицы видно, что наиболее подходящей является нотация BPMN, которая легка в изучении, удобна в создании моделей и имеет функциональный подход к проектированию. UML и ARIS имеют сложную панель управления, а также чрезвычайно трудны для восприятия и изучения. Проанализировав и выбрав технологию концептуального моделирования, перейдем к созданию модели бизнес-процесса «КАК ЕСТЬ».

1.3.2 Разработка и анализ модели бизнес-процесса управления заказами североамериканской телекоммуникационной компании «КАК ЕСТЬ»

Основная задача диаграммы «КАК ЕСТЬ» заключается в определении слабых мест бизнес-процесса и исходной точки для его изменения.

Модель бизнес-процесса «КАК ЕСТЬ» описывает принципы и механизмы неавтоматизированного функционирования бизнес-процесса.

Подобная информация является основой для комплексного, системного анализа процессов, поиска проблем и путей их преодоления.

На рисунке 1.4 представлена модель «КАК ЕСТЬ» с точки зрения менеджера по работе с клиентами, построенная по методологии BPMN.

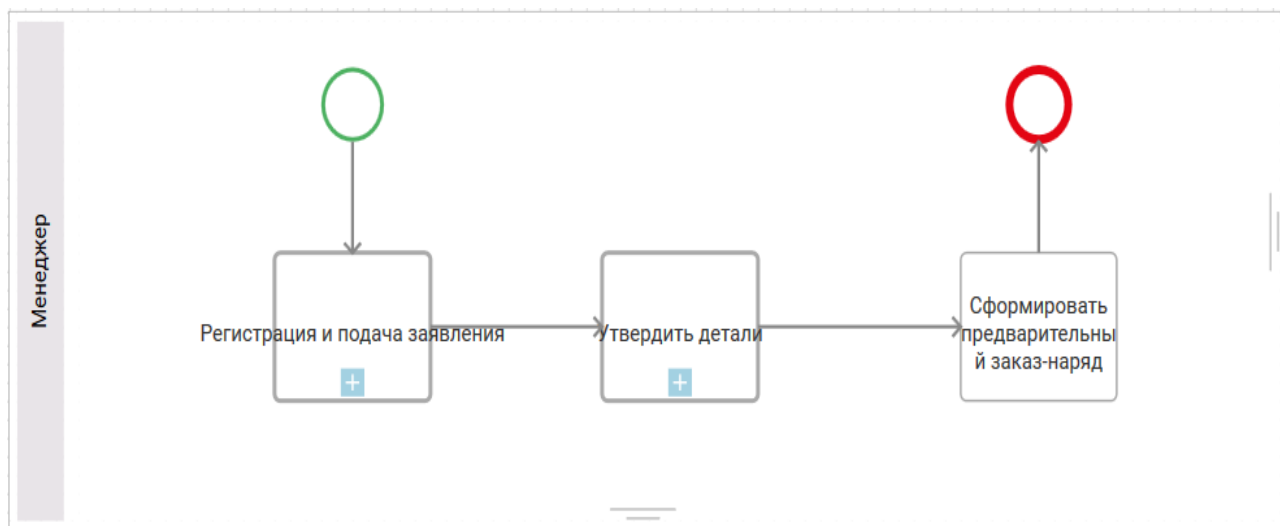


Рисунок 1.4 - Контекстная диаграмма «КАК ЕСТЬ» бизнес-процессов управления заказами клиентов в методологии BPMN

Первым делом клиент приходит в офис и его регистрирует менеджер. Затем клиент пишет заявление на подключение услуги. Если заявление оформлено правильно, клиент с менеджером утверждают детали, такие как:

- определение объема и срока выполнения заказанных работ;
- определение предварительной стоимости;
- согласование даты.

В конечном итоге, если все детали были утверждены с клиентом, менеджер переходит к оформлению предварительного заказ-наряда. Для более

подробного рассмотрения – декомпозируем все бизнес-процессы на подпроцессы. Декомпозиция регистрации клиента приведена на рисунке 1.5.

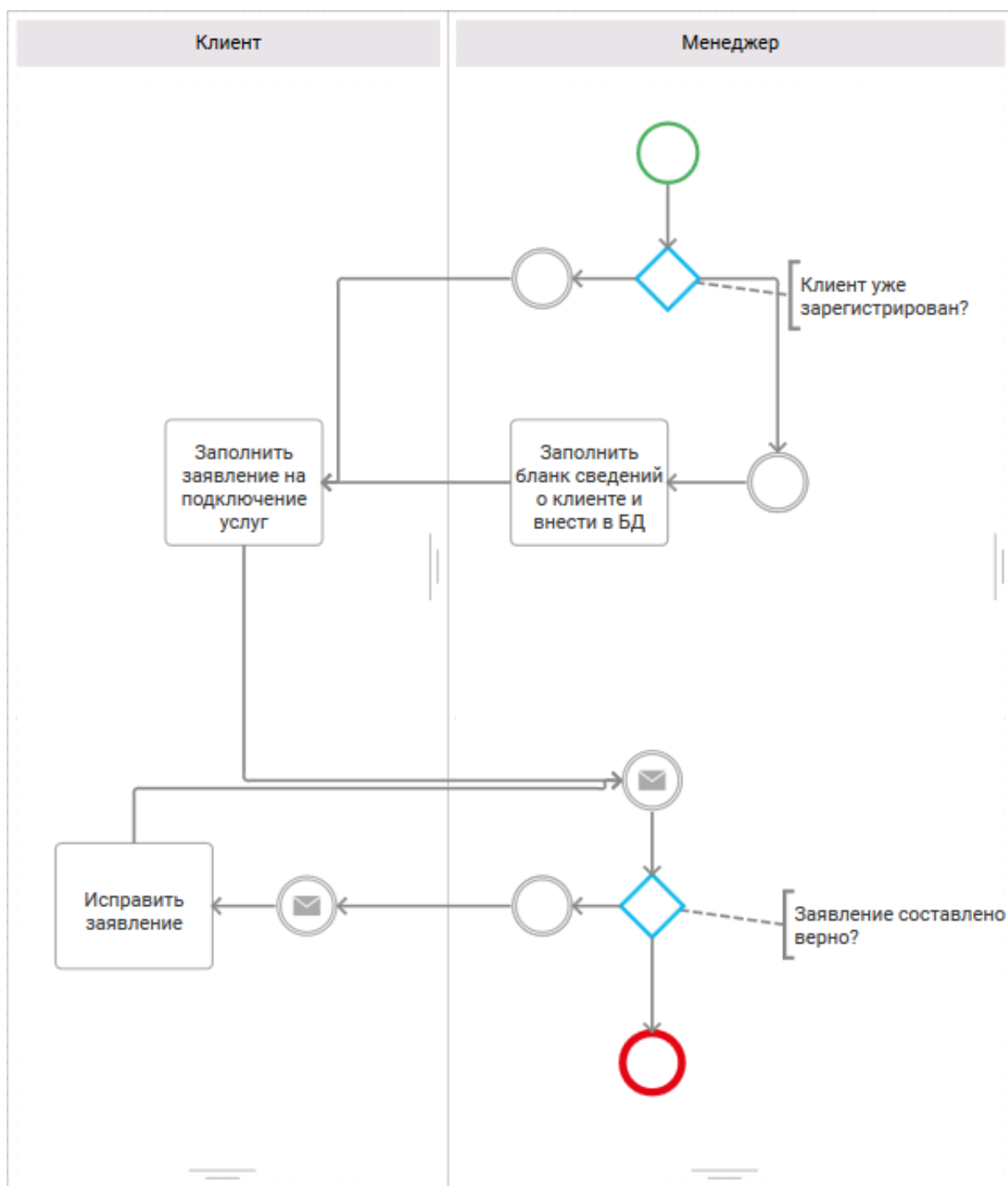


Рисунок 1.5 – Декомпозиция бизнес-процесса «Зарегистрировать клиента»

Декомпозиция утверждения деталей приведена на рисунке 1.6.

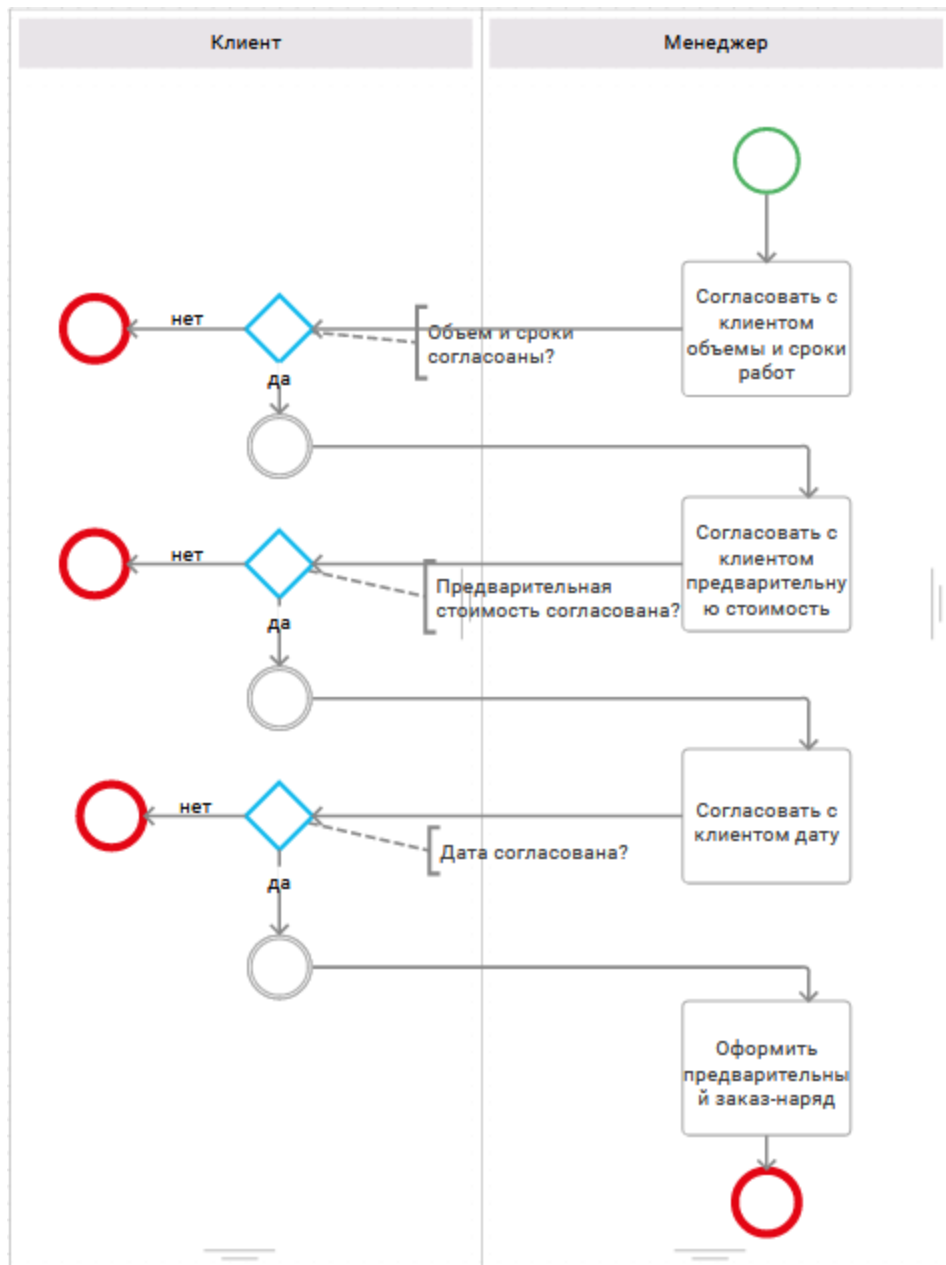


Рисунок 1.6 – Декомпозиция бизнес-процесса «Утвердить детали»

Анализ показал, что на эти бизнес-процессы тратится очень много лишних ресурсов, такие как время и кадры. Автоматизация оптимизирует затраты ресурсов.

После проведённого анализа процесса «Управление заказами» деятельности менеджера по работе с клиентами, используя структурное моделирование, перейдем к формированию требований к проектируемой информационной системе.

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

FURPS+ - классификация требований к программным системам, расширенная версия классификации требований FURPS, которая расшифровывается следующим образом:

- functionality – функциональные требования;
- usability – требования к удобству использования;
- reliability – требования к надежности;
- performance – требования к производительности;
- supportability – требования к поддержке.

Дополнительно включает в себя ограничения, разделенные на следующие группы требований:

- design – ограничения проектирования;
- implementation – ограничения разработки;
- interface – ограничения на интерфейсы;
- physical – физические ограничения.

Процессы «Заполнить заявление на подключение услуг» и «Заполнить бланк сведений о клиенте и внести в БД», входящие в состав бизнес-процесса «Зарегистрировать клиента», обладают следующими недостатками:

1. Долгий и ненадежный процесс формирования документов клиентом и менеджером по работе с клиентами. Данный недостаток может быть устранен путем автозаполнения документа при помощи добавления в проектируемую информационную систему соответствующих функций.

2. Отсутствие автоматизированного добавления, поиска клиентов. Данный недостаток может быть устранен путем добавления в информационную систему функций вывода всех клиентов по фамилии клиента, которые хранятся в базе данных.

Таблица 1.2 – Требования к системе

ИД	Требование	Статус	Полезность	Риск	Стабильность
Функциональные требования					
1	Регистрация клиента	Одобренные	Критичное	Средний	Низкая
2	Редактирование данных клиента	Одобренные	Критичное	Средний	Низкая
3	Добавление новой заявки	Одобренные	Критичное	Средний	Низкая
4	Отслеживание заявки по статусу	Одобренные	Критичное	Средний	Низкая
5	Формирование заказ-наряда	Одобренные	Критичное	Средний	Низкая
Требования к удобству использования					
6	Интерфейс пользователя должен быть логичен, прост и понятен в использовании	Одобренные	Критичное	Низкий	Низкая
7	Единый стиль и цветовая гамма	Одобренные	Критичное	Низкий	Низкая
8	Обучение и квалификация сотрудников	Одобренные	Критичное	Низкий	Низкая
9	Проверка введенных данных на корректность	Одобренные	Критичное	Низкий	Низкая
Требования к надёжности					
10	Разграничение прав доступа сотрудников и клиентов	Одобренные	Критичное	Средний	Средняя
11	Доступ к системе – 24 часа в сутки	Одобренные	Критичное	Низкий	Низкая
Требования к производительности					
12	Время отклика системы не должно превышать более чем 1 секунды	Предложенные	Критичное	Средний	Средняя

ID	Требование	Статус	Полезность	Риск	Стабильность
13	Расширение дополнительного функционала	Предложенные	Критичное	Средний	Средняя
14	Поддержка работоспособности - обновление данных	Предложенные	Критичное	Средний	Средняя
Ограничения					
Ограничения проектирования					
18	Средства разработки – документация должна быть в MS Word	Одобренные	Критичное	Низкий	Низкая
Ограничения реализации					
19	Используемая СУБД – MySQL	Одобренные	Критичное	Средний	Средний
20	Реализация бизнес-логики на Java Spring	Одобренные	Критичное	Средний	Средняя

После того, как выявлены все недостатки и требования, которые представлены в таблице 1.2, далее необходимо произвести анализ уже существующих аналогов.

1.4 Анализ существующих разработок на предмет соответствия сформулированным требованиям

Рассмотрим существующие аналоги CRM-систем. Проведем анализ и определимся с предложенными элементами информационной системы. Это необходимо для определения преимуществ, для того, чтобы выбрать направление проектирования. Далее будут представлены аналоги CRM-систем.

1. Битрикс24, пользовательский интерфейс изображен на рисунке 1.7.

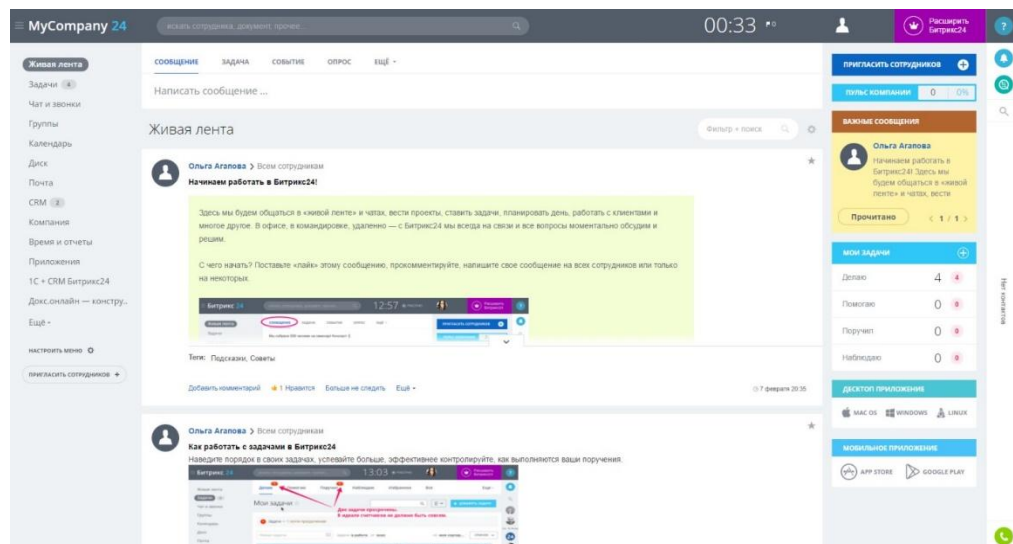


Рисунок 1.7 - Битрикс24

Воспользовавшись пробной версией, с которой можно ознакомиться по адресу <https://bitrix24.ru/> Представляет собой информационную систему для организации работы компании.

Имеет ряд преимуществ, таких как:

- облачное хранение данных;
- многофункциональное управление продажами;
- IP-телефония;
- мобильность.

2. CRM-линейка bmp'online, пользовательский интерфейс изображен на рисунке 1.8.

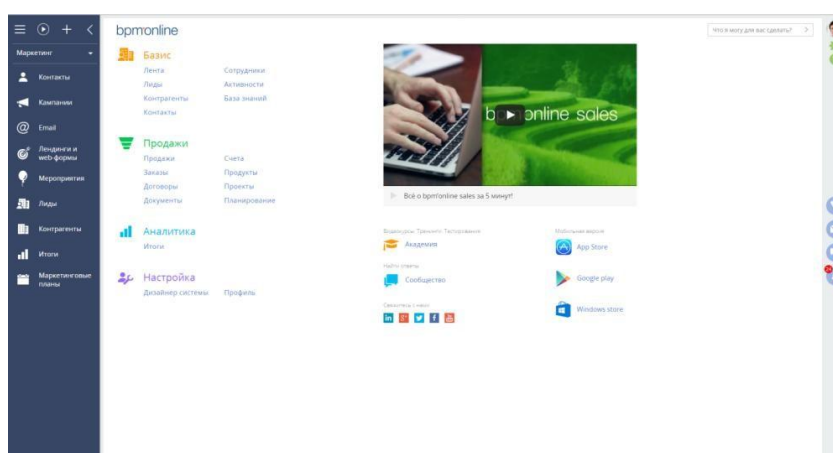


Рисунок 1.8 - CRM-линейка bmp'online

CRM-продукты на платформе bpm'online помогают компаниям управлять отношениями с клиентом: от привлечения до обслуживания и повторных продаж, так же система имеет множество удобных функций:

- многофункциональный документооборот;
- удобное управление заказами;
- возможность управление сервисом;
- интуитивно-понятный интерфейс;
- возможность создавать и развивать маркетинговые планы.

Представим в таблице 1.3 достоинства и недостатки рассмотренных CRM-систем.

Таблица 1.3 – Сравнение аналогичных информационных систем

№	Категории сравнения	Битрикс24	CRM-линейка bpm'online
1.	Интуитивно-понятный интерфейс	+	+
2.	Многофункциональный документооборот	-	+
3.	IP-телефония	+	-
4.	Мобильность	+	+
5.	Планирование проекта и управление задачами	+	+
6.	Облачное хранение данных	+	-
7.	Представление итогов, успешность сделок и продаж на графиках	+	+
Итого:		6/7	5/7

Как видно из таблицы 1.3, рассмотренные выше CRM-системы не полностью удовлетворяют необходимым требованиям. Так как данные системы имеют специфичный набор функций было принято решение о создании CRM-системы для компании, удовлетворяющей требованиям заказчика.

1.5 Постановка задачи на разработку проекта создания/внедрения автоматизированной информационной системы

Разрабатываемая информационная система с элементами CRM-системы будет внедрена в основной проект североамериканской телекоммуникационной компании, для того что бы работа менеджера по работе с клиентами стала более эффективной и быстрой. Цель создания автоматизированной информационной системы является обеспечение качественного и надежного выполнения учета и формирования документов, а также простое и удобное в использовании отслеживание заявок. Анализ существующих аналогов CRM-систем, рассматриваемых в пункте 1.4, показал определенные недостатки информационных систем, поэтому для достижения поставленной цели создания информационной системы с элементами CRM-системы были сформированы основные требования:

- введение базы данных, которая содержит всю информацию о клиенте и его контактах, которые помогут осуществить связь сотрудника с клиентом;
- интуитивно-понятный интерфейс для сотрудников компании, который поможет быстро разобраться с функционалом информационной системы, так же менеджер по работе с клиентами будет четко понимать, что произойдет после того, как он заполнит форму. Часть функционала сделать иконками, которые всем понятны и знакомы, чем размещать ссылки;
- автозаполнение шаблонов документов позволит исключить появление ошибки и сократит время на рутинное заполнение данными документов;
- централизованное хранение всех документов в базе данных, просмотр в любой момент времени.

Сотрудники североамериканской телекоммуникационной компании тратят много времени и сил на выполнение лишних задач. Именно поэтому необходима автоматизированная информационная система с элементами CRM-системы, которая позволит сократить время и обеспечит более качественное и надежное выполнение процессов.

1.6 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»

При создании контекстной диаграммы «КАК ДОЛЖНО БЫТЬ» (TO-BE), представленной на рисунке 1.9, внедряется исполнитель «Автоматизированная информационная система», который отвечает за автоматизацию бизнес-процессов регистрации, формирования и обработки заявок.

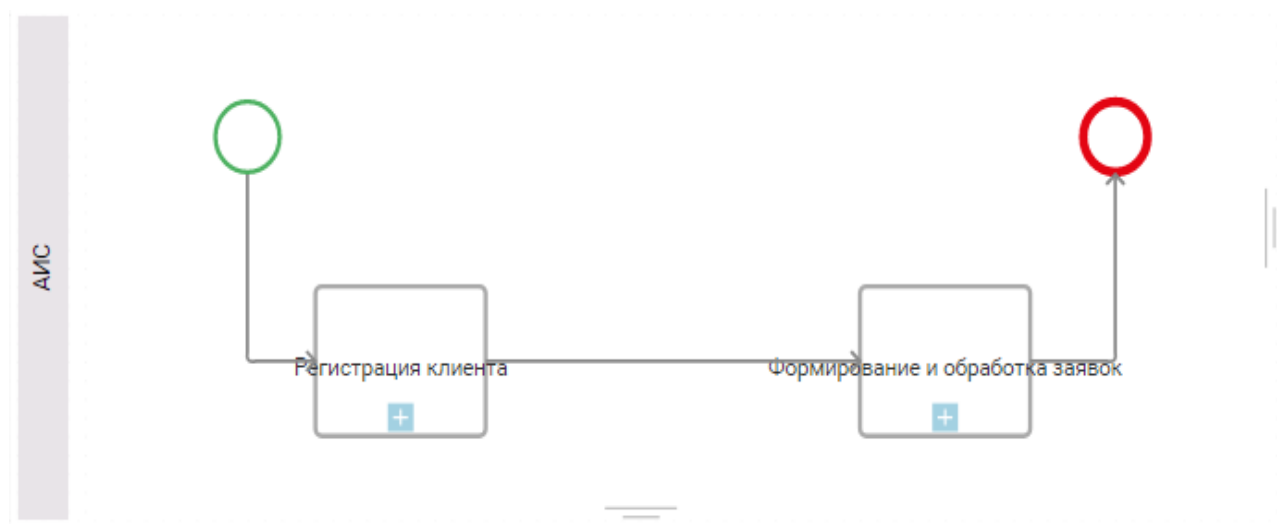


Рисунок 1.9 - Контекстная диаграмма «КАК ДОЛЖНО БЫТЬ» бизнес-процессов

Далее следует посмотреть на эти процессы более подробно, для чего проведем декомпозицию этих бизнес-процессов. Результат декомпозиции «Регистрация клиента» представлен на рисунке 1.10. На декомпозиции видно, что большинство рутинных процессов перешло под управление автоматизированной информационной системы.

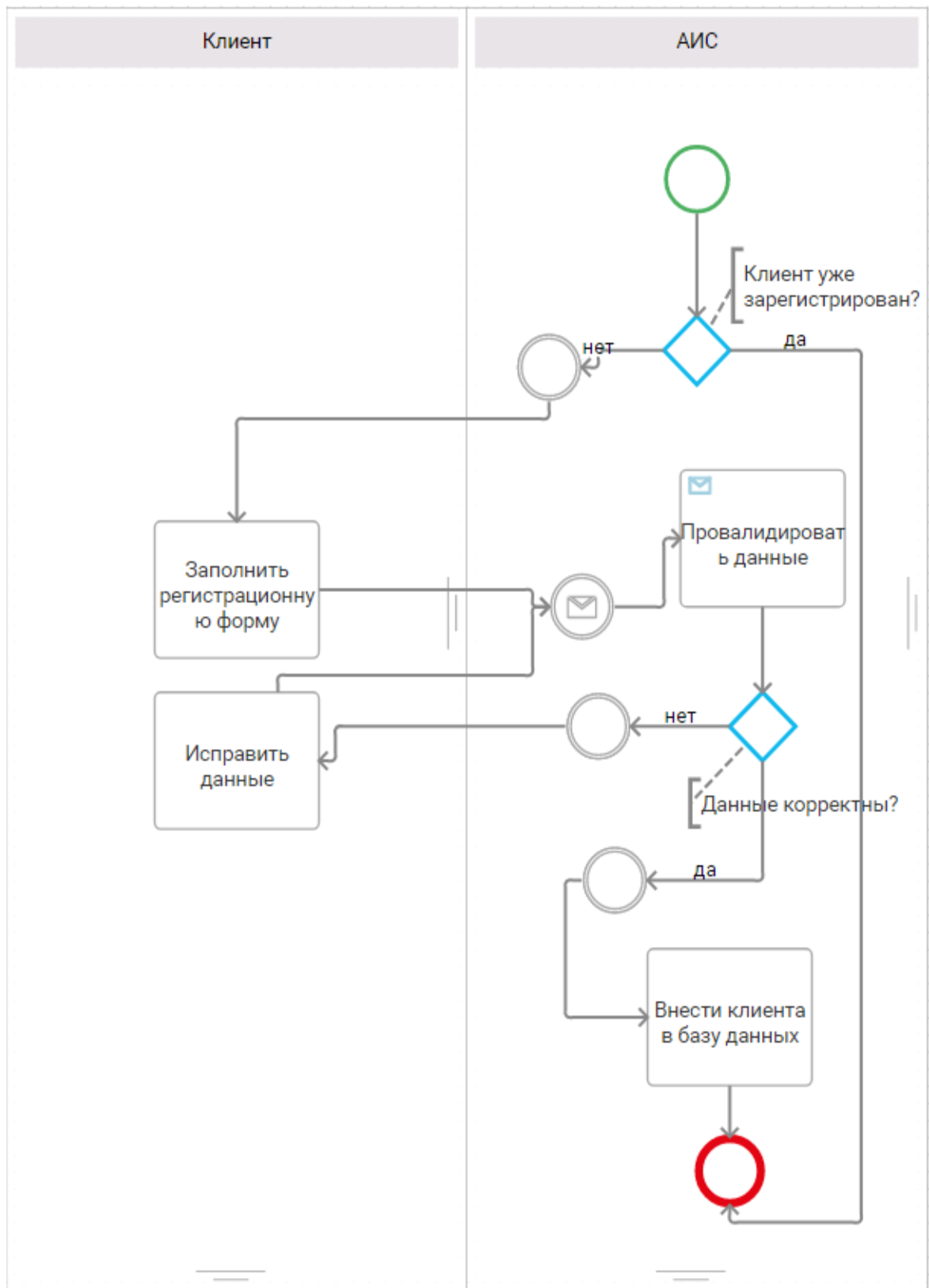


Рисунок 1.10 – Декомпозиция бизнес-процесса «Регистрация клиента»

Результат декомпозиции «Формирование и обработка заявок» представлен на рисунке 1.11.

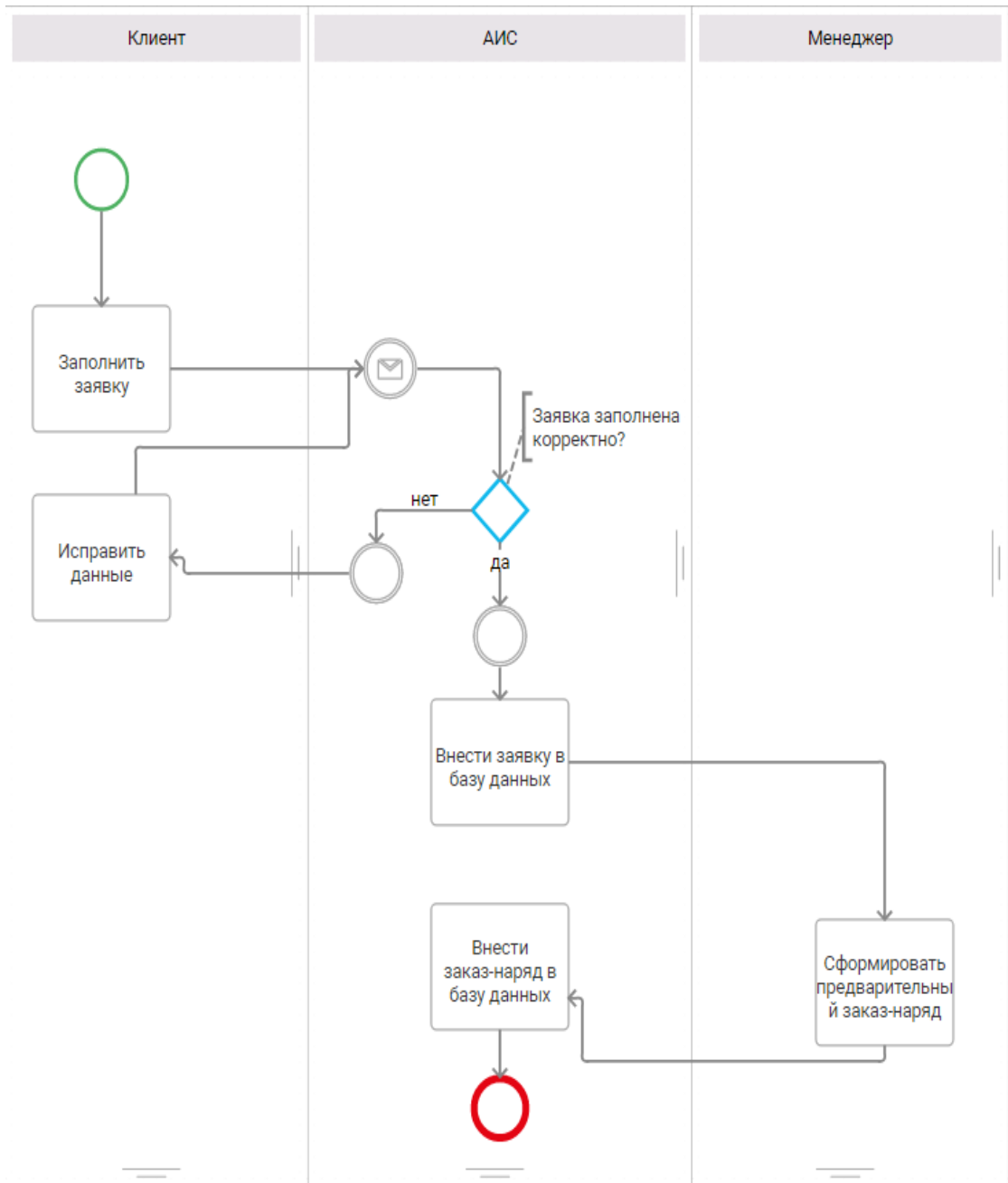


Рисунок 1.11 – Декомпозиция бизнес-процесса «Формирование и обработка заявок»

После автоматизации бизнес-процессов путем внедрения автоматизированной информационной системы взаимодействие с клиентами проходит гораздо более быстро и продуктивно. Большинство процессов перешло под управление АИС. С помощью автоматизированной информационной системы с элементами CRM-системы сотрудники компании

могут организовать, создавать и управлять запросами клиентов, а также отслеживать все заявки.

Выводы по главе 1

В первой главе была выбрана технология концептуального моделирования, построена диаграмма основных процессов деятельности менеджера по работе с клиентами североамериканской телекоммуникационной компании. Также был проведен анализ существующих аналогов CRM-систем, поставлена цель разработки и выделены основные задачи, которые послужили основанием для необходимости разработки автоматизированной информационной системы с элементами CRM-системы для компании.

Глава 2 Логическое проектирование CRM-системы для североамериканской телекоммуникационной компании

2.1 Выбор технологий логического моделирования автоматизированной информационной системы управления заказами североамериканской телекоммуникационной компании

Следующим этапом проектирования является логическое моделирование, которое представляет собой процедуру проверки функционирования логической схемы, основанной на конкретной модели данных. Методология реализуется через конкретные технологии и поддерживающие их стандарты, методики.

CASE – автоматизированная разработка программного обеспечения. CASE-средства – представляет собой набор инструментов и методов для проектирования программного продукта.

CASE-средства обеспечивают процесс разработки и помогают создавать практически готовый к применению продукт. Применяя эту технологию, сокращается время разработки и количество ошибок, а также повышается эффективность работы. CASE-средства представляют собой набор инструментальных средств, которые позволяют моделировать и анализировать предметную область на всех этапах разработки.

Логическая модель автоматизированной информационной системы будет описана при помощи диаграммы вариантов использования и диаграммы классов. Логическая и физическая модели будущей информационной системы будут построены с помощью языка моделирования UML.

StarUML - это пакет с открытым исходным кодом для быстрого, гибкого, расширяемого, функционального UML-моделирования. StarUML поддерживает одиннадцать видов диаграмм и MDA (Модельно-управляемая архитектура), реализуя концепцию профилей UML. В соответствии с требованиями пользователя среда разработки быстро и легко настраивается и имеет широкий спектр функциональных возможностей.

Перейдем к созданию логической модели автоматизированной информационной системы.

2.2 Логическая модель информационной системы и её описание

Разработка логической модели предполагает переход от структурной диаграммы «КАК ДОЛЖНО БЫТЬ» к диаграмме вариантов использования, отражающей функциональный аспект логической модели системы, а также разработку диаграммы классов предметной области рассматривающая элементный аспект логической модели системы.

Разработанная диаграмма вариантов использования представлена на рисунке 2.1.

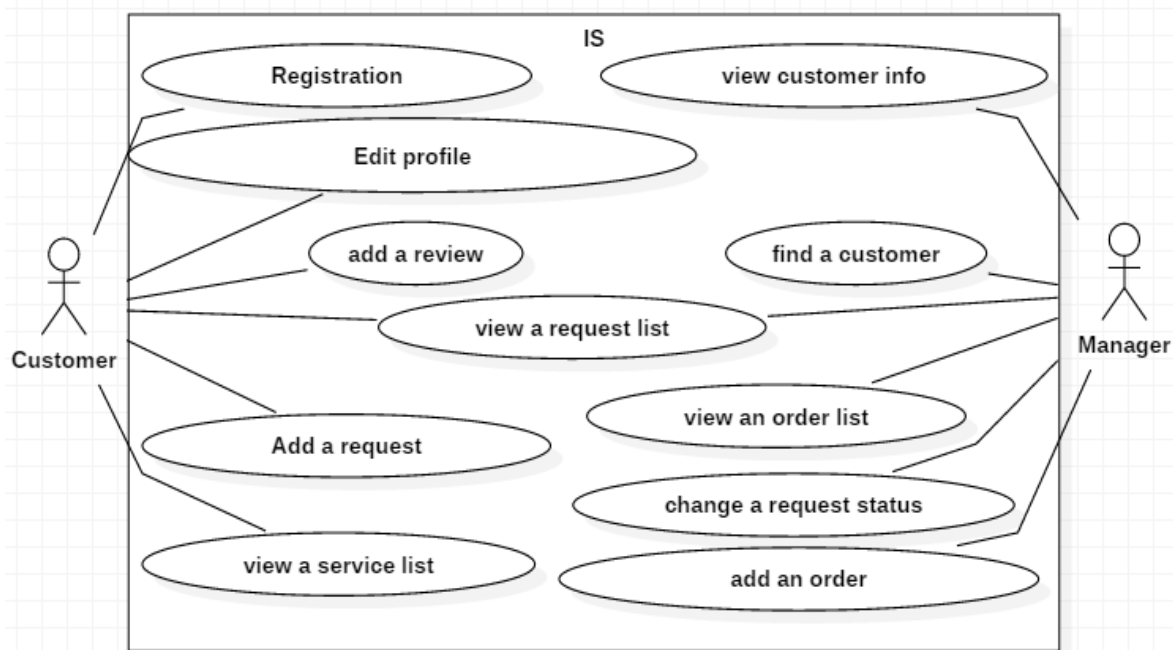


Рисунок 2.1 – Диаграмма вариантов использования

На диаграмме представлены следующие роли:

- клиент - лицо, использующее услуги компании, которое предоставляет сотруднику свои контакты и данные о заявке;
- менеджер - должностное лицо, ответственное за консультацию клиентов, составление документов, оформление и отслеживание заявок, просмотр и ведение клиентской базы данных.

Краткая характеристика прецедентов диаграммы представлена в таблице 2.1.

Таблица 2.1 - Характеристика прецедентов

Прецедент	Характеристика
Регистрация в системе	Регистрация в системе нового клиента.
Редактирование данных профиля	Изменение данных клиента.
Оформление заявлений	Оформление заявлений на услугу.
Отслеживание заявок	Просмотр заявок и их статус выполнения.
Просмотр данных клиента	Просмотр данных клиента в системе.
Просмотр списка услуг	Просмотр всех доступных клиенту услуг.
Формирование заказ-наряда	Оформление заказ-наряда.
Изменение статуса заявки	Изменение статуса заявки менеджером.
Просмотр всех заявок	Просмотр всех заявок в системе.
Поиск клиента	Поиск данных клиента в системе по его логину.
Добавление отзыва/предложения	Добавление отзыва клиентом или анонимным пользователем с возможностью обратного звонка.

Разработанная диаграмма классов показывает атрибуты и операторы классов и их взаимосвязь, а также описывает структуру. Данная диаграмма представлена на рисунке 2.2.

В диаграмме были использованы такие связи как: ассоциация, агрегация и композиция. Ассоциация показывает, что объекты одной сущности (класса) связаны с объектами другой сущности таким образом, что можно перемещаться от объектов одного класса к другому. Агрегация - это разновидность ассоциации при отношении между целым и его частями. Как тип ассоциации агрегация может быть именованной. Одно отношение агрегации не может

включать более двух классов (контейнер и содержимое). Композиция - более строгий вариант агрегации. Известна также как агрегация по значению.

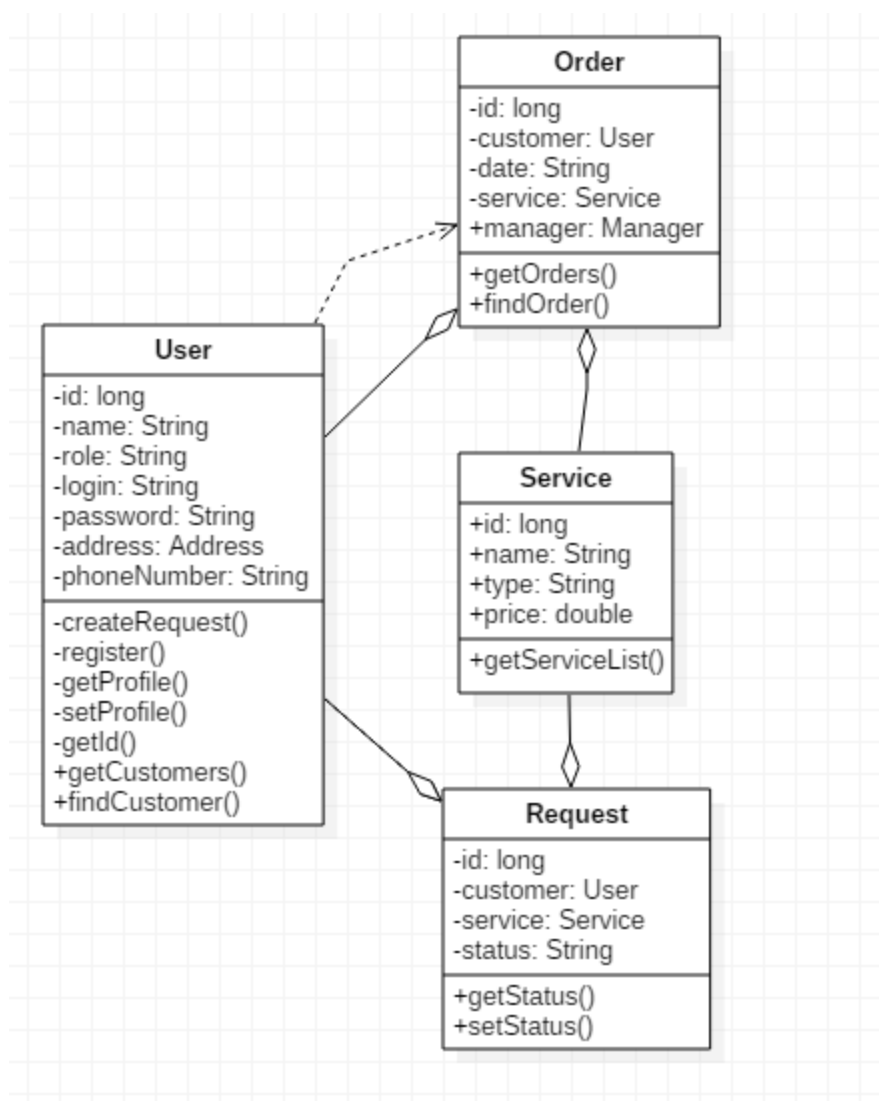


Рисунок 2.2 – Диаграмма классов

Так же на рисунке 2.3 изображена диаграмма последовательности, которая отображает процесс регистрации клиента, подачи заявления и формирование предварительного заказ-наряда.

Взаимодействие между объектами и субъектами происходит следующим образом:

- 1) клиент регистрируется в системе;
- 2) в системе формируется заявление, данные в которую вносит клиент;

- 3) менеджер просматривает заполненные заявления и передает данные необходимые для последующей подготовки;
- 4) после того как данные были утверждены, менеджер заполняет предварительный заказ-наряд в системе.



Рисунок 2.3 – Диаграмма последовательности

Таким образом, с помощью диаграммы последовательности CRM-системы для североамериканской телекоммуникационной компании, была проверена логика функционирования системы и описана последовательность работы.

Далее построим концептуальную и логическую модель будущей базы данных, с помощью вышеописанных диаграммой вариантов использования и диаграммой классов.

2.3 Разработка концептуальной и логической моделей данных информационной системы компании

Концептуальная модель данных предполагает описание главных сущностей и их отношения.

Построение этой модели, изображенной на рисунке 2.4, является важной частью при проектировании. Концептуальная модель данных имеет пять сущностей: «Заявка», «Пользователь», «Заказ», «Сервис» и «Помощь». Далее построим логическую модель данных, которая расширяет концептуальную, определяя атрибуты для сущностей.

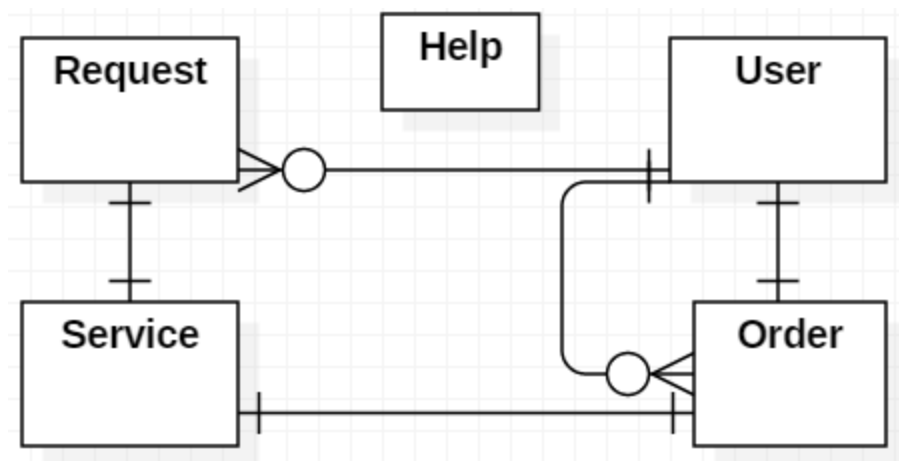


Рисунок 2.4 – Концептуальная модель данных

Логическая модель изображена на рисунке 2.5.

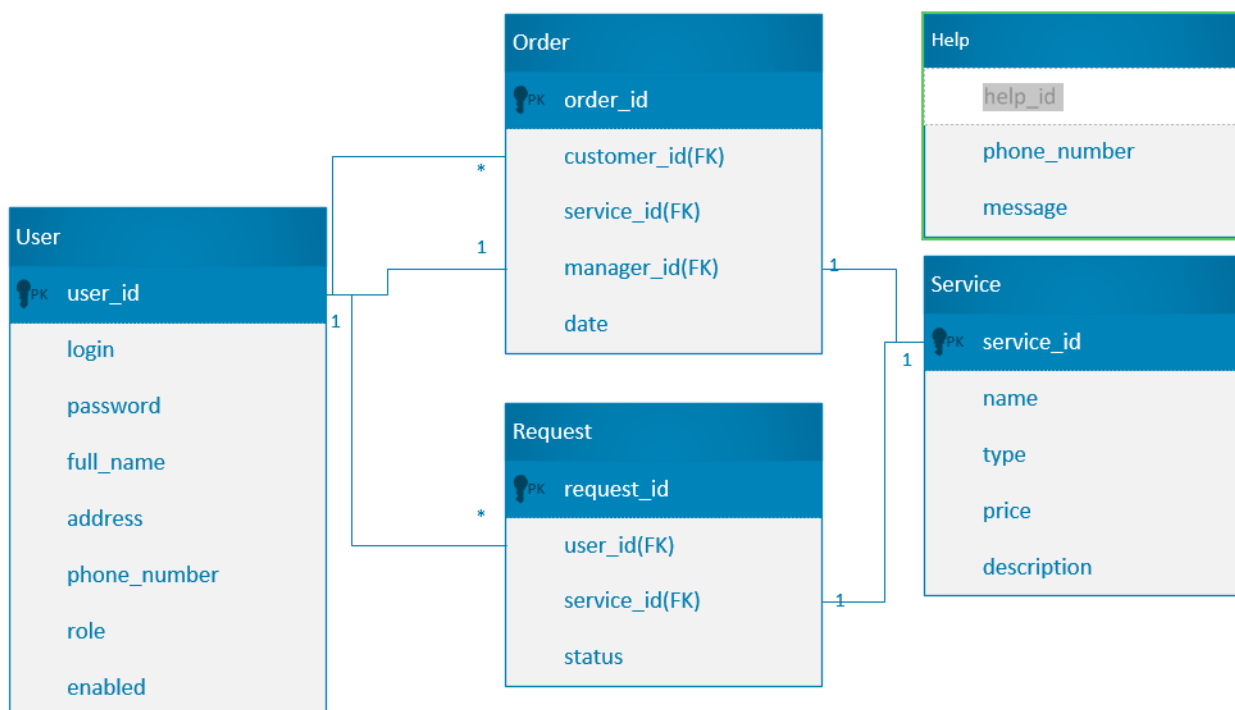


Рисунок 2.5 - Логическая модель данных

Опишем подробно сущности «Заявка», «Пользователь», «Заказ», «Сервис» и «Помощь», представленные на логической модели:

- сущность «Заявка» содержит данные о заявке, клиенте, сервисе и статус. По умолчанию стоит «В обработке», то есть заявка была принята;
- сущность «Пользователь» содержит данные о пользователе, который в зависимости от роли может являться и клиентом и менеджером, данные и контакты – ФИО, адрес, телефон, а также логин и пароль для входа в систему;
- сущность «Заказ» содержит данные о заказе, клиенте, которому предоставляется сервис, сам сервис и менеджер, который оформил заказ-наряд;
- сущность «Сервис» содержит данные об предоставляемой компанией услуге – название услуги и ее цена;
- сущность «Помощь» содержит данные об обратной связи. Клиент или анонимный пользователь может оставить обращение и телефон для обратной связи.

После того, как были описаны сущности и построена логическая модель, перейдем к требованиям аппаратно-программного обеспечения информационной системы.

2.4 Требования к аппаратно-программному обеспечению информационной системы

Техническое обеспечение системы, а также требования к безопасности и производительности должны эффективно выполняться, чтобы обеспечить качество работы и использовать существующие на предприятии технические средства.

Аппаратно-программное обеспечение должно соответствовать следующим требованиям:

- операционная система – Windows 7 и выше;
- центральный процессор – частота не менее 2 ГГц;
- оперативная память – 2 GB и выше;
- свободное место на жестком диске – 500Мб и больше;

- подключение к сети интернет;
- офисные пакеты – Microsoft Word 2007 и выше.

Серверная часть:

- работа 24 часа;
- процессор – не ниже Pentium IV 1ГГц;
- оперативная память – от 1 Гб;
- жесткий диск – от 1 Гб.

Требования к серверу СУБД и АИС:

- процессор – частота не менее 2.4 ГГц;
- оперативная память – не менее 4 Гб;
- свободное место на жестком диске - не менее 1 Гб;
- подключение к сети Интернет;
- операционная система - Windows.

Требования к безопасности: требуется защитить исходный код общей части информационной системы. Требуется разграничение доступа. Пароли сотрудников и клиентом хранятся в зашифрованном виде. На уровне СУБД должно быть реализовано разграничение доступа к данным в БД.

Выводы по главе 2

Глава 2 описывает логическое проектирование информационной системы. В ней происходит выбор технологии логического моделирования, а также проектирование БД, происходит построение логической модели. Описывает все выявленные требования к аппаратно-программному обеспечению автоматизированной информационной системы.

Глава 3 Практическая реализация CRM-системы

3.1 Выбор технологии разработки программного обеспечения

Важнейшим этапом при разработке CRM-системы является выбор технологии реализации. Для правильной реализации CRM-системы для североамериканской телекоммуникационной компании можно выделить несколько языков программирования: Python, Ruby, Java.

Python - универсальный язык программирования, предназначен для работы с базами данных и обработки текстов. Используется для программирования Internet и Web приложений.

Ruby - объектно-ориентированный язык программирования, предназначен для разработки Web приложений.

Java - объектно-ориентированный язык программирования, который способен создавать динамические Web сайты, а так же работать с базами данных.

Для правильного выбора языка программирования проведём сравнительный анализ трёх языков программирования, который представлен в таблице 3.1

Таблица 3.1 - Сравнительный анализ языков программирования

Критерий	Языки программирования		
	Python	Ruby	Java
Поддержка большим количеством серверов	+	-	+
Простота синтаксиса	-	-	+
Эффективность	+	-	+
Быстродействие	-	+	+
Обучаемость	+	+	+
Широкие возможности с базами данных	-	-	+
Потребление памяти	-	+	-
Наличие опыта работы с языком программирования	-	-	+
Итого	3	3	8

По итогам сравнительного анализа языков программирования был выбран язык программирования Java, так как данный язык имеет широкие возможности для работы с базами данных, а также поддерживается большим количеством серверов.

3.2 Описание основного принципа работы CRM-системы для североамериканской телекоммуникационной компании

Данная система позволяет клиентам регистрироваться и авторизоваться в системе. Оставлять отзывы, просматривать список услуг, оставлять заявку на услугу, редактировать профиль. Так же система позволяет менеджеру просматривать и изменять статус заявки, просматривать список клиентов и производит поиск по их логину, формировать заказ-наряд.

Рассмотрим некоторые функции системы от лица клиента. При запуске системы появляется главное меню с возможностью пройти аутентификацию и авторизацию в системе. Изображено на рисунке 3.1.

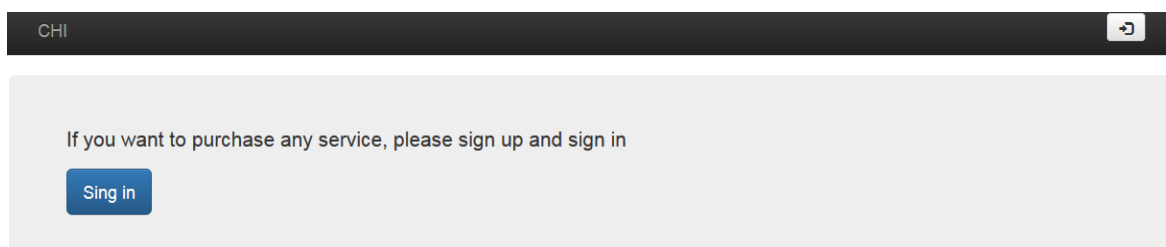


Рисунок 3.1 – Главная страница

При нажатии на кнопку «Sign in» отображается форма (Рис. 3.2) для регистрации и аутентификации/авторизации пользователя (Приложение А).

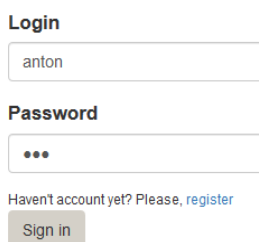


Рисунок 3.2 – Форма аутентификации и авторизации пользователя

После того, как пользователь введет свои данные и войдет в систему ему станут доступны такие функции системы как:

- просмотр сервисов;
- оставить заявку;
- оставить отзыв с возможностью обратной связи;
- просмотр заявок;
- просмотр и редактирование профиля.

На рисунке 3.3 изображена страница для просмотра сервисов (Приложение Б).

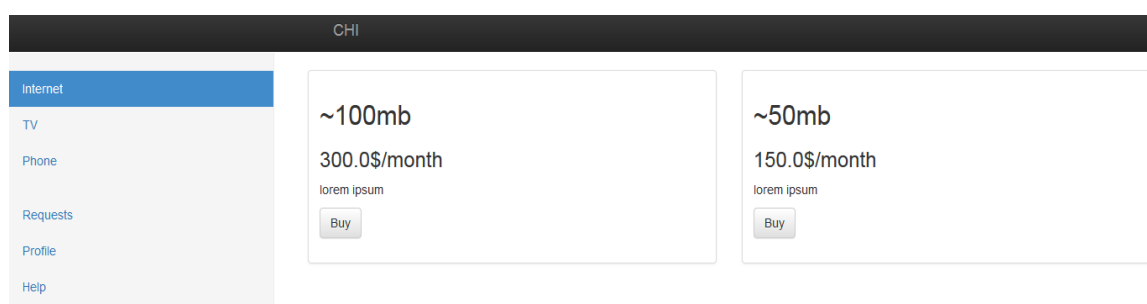


Рисунок 3.3 – Просмотр сервисов

При нажатии на кнопку «Buy» пользователь попадает на форму с подачей заявки. Форма для подачи заявки (Приложение В) изображена на рисунке 3.4. Все данные форма получает автоматически из выбранного сервиса и данных пользователя. Если у пользователя не будет заполнено поле адрес или поле телефон в профиле – подача заявки будет недоступна.

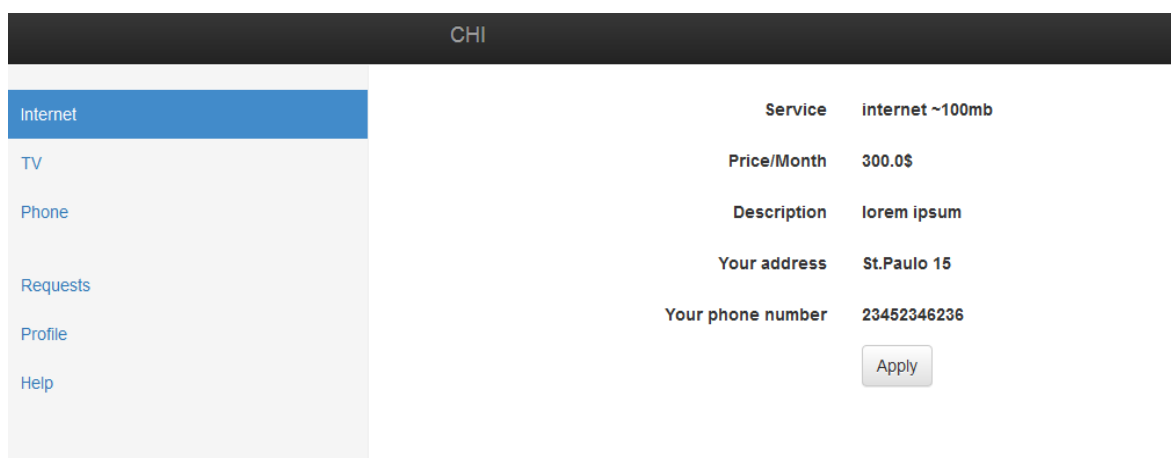
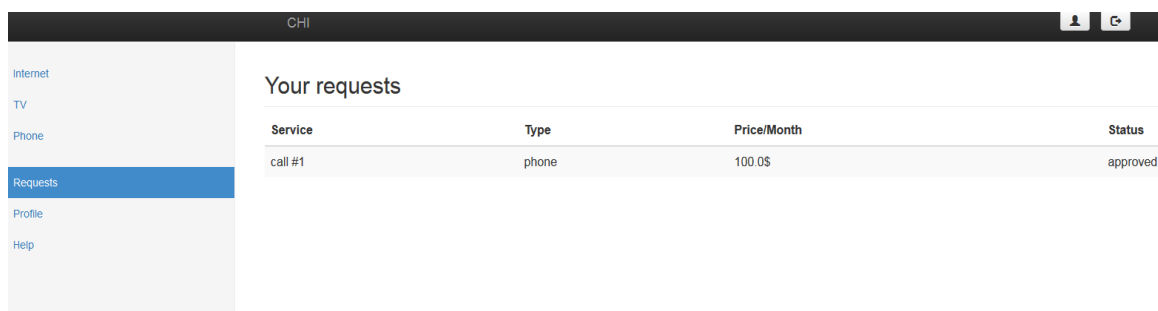


Рисунок 3.4 – Добавление заявки

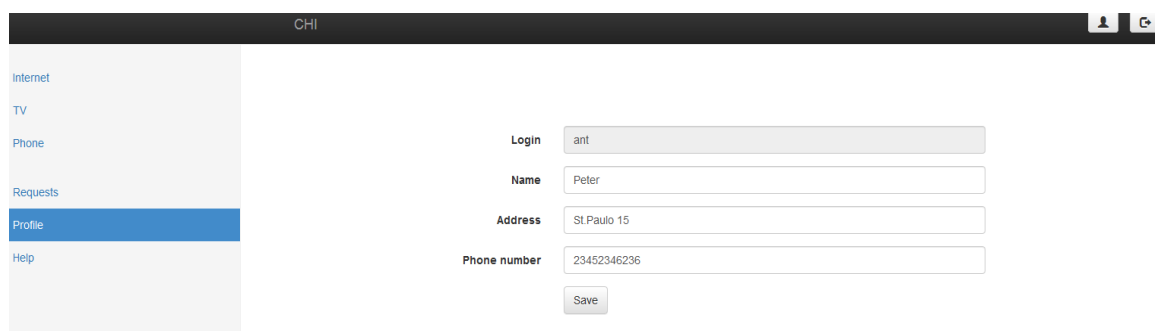
Так же клиент может просматривать поданные заявки на странице (Рис. 3.5), где он может наблюдать изменения статуса заявки (Приложение Г).



Service	Type	Price/Month	Status
call #1	phone	100.0\$	approved

Рисунок 3.5 – Просмотр списка поданных заявок

Клиенту так же доступна страница редактирования своего профиля (Рис. 3.6).



CHI

Internet
TV
Phone
Requests
Profile
Help

Login: ant

Name: Peter

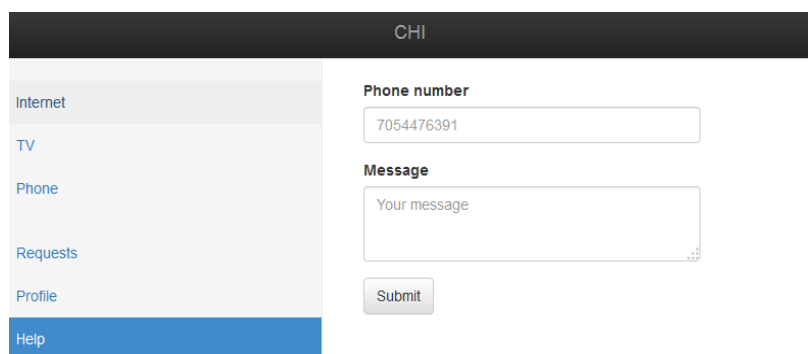
Address: St. Paulo 15

Phone number: 23452346236

Save

Рисунок 3.6 – Просмотр и редактирование профиля

Если клиент хочет оставить отзыв или предложение он может это сделать на форме (Рис. 3.7) для отправки отзывов и предложений с возможностью обратной связи.



CHI

Internet
TV
Phone
Requests
Profile
Help

Phone number: 7054476391

Message: Your message

Submit

Рисунок 3.7 – Форма для добавления отзыва и предложений с возможностью обратной связи

В ходе работы была спроектирована информационная система управления заказами, в которой были решены основные функциональные задачи, а также реализовано удобство пользования. Во время описания основного принципа работы CRM-системы для североамериканской телекоммуникационной компании, был представлен алгоритм приёма заявок на установку телекоммуникационной услуги. По итогам описания основного принципа работы CRM-системы для североамериканской телекоммуникационной компании, можно сделать вывод, что разработанная CRM-система работоспособна и удовлетворяет заданным требованиям.

3.3 Тестирование CRM-системы для североамериканской телекоммуникационной компании

Тестирование программного продукта - это процесс проверки программного продукта на предмет ошибок и соответствия требованиям.

В качестве метода тестирования будет использоваться метод «Чёрного ящика (Black box testing)». Тестирование черного ящика - это проверка продукта согласно предъявляемым требованиям без изучения программного кода и без доступа к базе данных. Для проведения теста, необходимо составить тест кейсы, которые представлены в таблице 3.2.

Таблица 3.2 - Тест-кейсы CRM-системы для североамериканской телекоммуникационной компании.

№	Название	Инструкции	Результат
1	Проверка наличия всех необходимых элементов и тестирование способности системы		
1.1	Укомплектованность главного меню	Проверить наличие необходимых элементов (Сервисы, Заявки, Профиль, Обратная связь)	Все необходимые элементы в наличии и готовы к работе.

№	Название	Инструкции	Результат
1.2	Укомплектованность меню для входа и регистрации	Проверить наличие необходимых элементов	Все необходимые элементы в наличии и готовы к работе.
1.3	Укомплектованность меню пользователя	Проверить наличие необходимых элементов (списки заявок, страница личного кабинет)	Все необходимые элементы в наличии и готовы к работе.
1.4	Формы для ввода данных	Проверить наличие форм для ввода данных (онлайн подача заявок, обратная связь)	Формы в наличии и готовы к заполнению
1.5	Наличие модуля входа/выхода в систему	Проверить наличие модуля входа/выхода в систему	Модуль в наличии и готов к работе
1.6	Наличие каждого модуля каталога сервисов	Проверить наличие всех модулей каталога сервисов	Модули в наличии и готовы к работе
2	Тестирование функциональных возможностей системы		
2.1	Соответствие элементов панели управления	Проверить соответствие элементов панели управления их функциональному модулю	Все элементы соответствуют своему функциональному модулю
2.2	Соответствие вывода информации о выбранном сервисе	Проверить соответствие вывода информации по выбранному сервису	Все данные выводятся правильно
2.3	Соответствие данных о заявках	Проверит соответствие вывода заполненных данных заявок	Все заполненные данные выводятся и записываются корректно
3	Тестирование бизнес-логики (импорт данных и т.д.)		
3.1	Работа системы без авторизации	Проверить возможность подачи заявки без регистрации	Без регистрации модуль подачи заявки доступен
3.2	Работа модуля редактирования заявки	Отредактировать заявку клиента, присвоить ей статус	Редактирование заявки доступно

№	Название	Инструкции	Результат
3.3	Вывод данных о заявке для клиента	Проверить вывод данных о заявке для клиента	Вывод производится корректно
3.4	Модуль отправки отзывов и предложений	Проверить отправку отзывов и предложений	Отзывы и предложения отправляются
3.5	Модуль изменения личных данных клиента	Проверить возможность изменения данных клиента	Данные редактируются
4	Тестирование дополнительных функций		
4.1	Выход из системы (выход из профиля)	Проверить выход из профиля	Выход из профиля происходит корректно
4.2	Сохранение сеанса пользователя	Проверка сохранения сеанса пользователя	Сеанс сохраняется

По итогам тестирования система доказала свою работоспособность и удовлетворила все выдвинутые технические требования.

Далее перейдём к проверке пригодности системы для использования на предприятии.

3.4 Обоснование эффективности использования CRM-системы для североамериканской телекоммуникационной компании

Основной целью при разработке CRM-системы для североамериканской телекоммуникационной компании было автоматизировать процессы управления клиентскими отношениями, в результате чего нагрузка на менеджера должна существенно снизиться.

Данная цель достигается посредством уменьшения работы менеджера непосредственно с клиентом. Снижение участия менеджера в работе с клиентом происходит посредством перекладывания части функций, а именно «Регистрация клиента», «Подача заявления», «Просмотр заявок» с менеджера на CRM-систему для североамериканской телекоммуникационной компании, также клиенту теперь доступна возможность оставить вопрос/отзыв с

возможностью обратного звонка, что в свою очередь, позволяет менеджеру по продажам грамотно распределить своё рабочее время, спланировать день.

Одно из основных достоинств для клиента является добавление возможности проверить свою заявку онлайн, это освобождает клиента от необходимости совершать звонок или спрашивать менеджера лично в офисе для уточнения статуса заявки. С точки зрения менеджера, представленное новшество освобождает менеджера от необходимости консультирования клиента о статусе заявки, теперь ему достаточно изменить статус заявки и в добавок может найти подробную информацию о клиенте или проверить информацию о заявке через специальную форму.

Далее приводится сравнение статистических данных в виде диаграмм. На рисунке 3.7 изображено сравнение среднего количества выполненных задач менеджером за день. Под выполненными задачами подразумеваются задачи, которые не связаны с клиентом, например, формирование заказ-наряда.

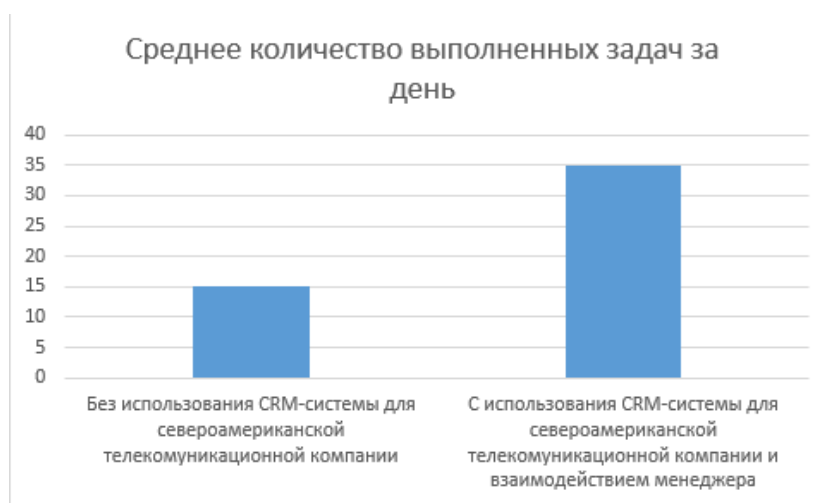


Рисунок 3.7 – Среднее количество выполненных задач за день

По результатам, количество выполненных задач повысилось более чем в 2 раза.

Следующим параметром, который будет представлен, является среднее количество обработанных заявок за день. Под обработанными заявками понимаются заявления на установку услуги из предложенных сервисов. Наглядное сравнение результатов приводится на рисунке 3.8



Рисунок 3.8 – Среднее количество обработанных заявок

Взяв за основу данные диаграммы 3.8 можно утверждать, что благодаря введению в компанию CRM-системы количество обработанных заявок увеличилось более, чем в два раза. Это происходит из того, что заявки регистрируются в системе, а это значит, что может быть подано и обработано несколько заявок одновременно без внимания и участия менеджера.

Одним из важных факторов оценки работоспособности CRM-системы для предприятия является время, которое необходимо для обработки одной заявки. Сравнение этого фактора показано на рисунке 3.9.



Рисунок 3.9 – Затраченное время на обработку одной заявки

После анализа количества обработанных заявок и времени, затраченного на одну заявку, для наглядности необходимо сравнить эти два фактора (рис. 3.10).



Рисунок 3.10 – Соотношение количества задач и затраченного времени

В итоге, можно сделать вывод, что разработка и дальнейшее использование системы полностью оправдывает себя и оптимизирует работу менеджера. Разработанная CRM-система для компании имеет возможности для расширения функциональности при необходимости.

Вывод по третьей главе

Была выбрана технология разработки CRM-системы для североамериканской телекоммуникационной компании. Также была реализована CRM-система для североамериканской телекоммуникационной компании, которая отвечает предъявленным требованиям заказчика.

Была описана работа системы, представлены экранные формы системы, раскрыты возможности системы.

Для проверки работоспособности системы было проведено тестирование системы методом «Чёрного ящика». В результате чего было доказано, что система является стабильной и исправной. После тестирования CRM-системы для североамериканской телекоммуникационной компании, была проведена проверка системы на её пригодность для использования на предприятии, в

результате данной проверки было доказано, что CRM-система для североамериканской телекоммуникационной компании может применяться для использования на предприятии и положительно влияет на работу менеджера по продажам. В итоге CRM-система для североамериканской телекоммуникационной компании была признана исправной и работоспособной.

Заключение

Итогом выпускной квалификационной работы является разработанная CRM-система для североамериканской телекоммуникационной компании. Данная система предназначена для автоматизации и управлением клиентскими отношениями. Организации онлайн подачи заявления и вывод данных о заявках клиента в режиме онлайн. Также разработанная система позволит осуществлять отправку отзывов и предложений клиента.

Во время проектирования CRM-системы для североамериканской телекоммуникационной компании был произведен анализ предметной области, на основании которого было принято решение о разработке CRM-системы. А также рассмотрены аналоги CRM систем которые представлены на рынке. В результате рассмотрения аналогов были выявлены слабые стороны конкурентов. Принимая во внимание анализ аналогов были сформулированы основные требования к CRM-системе для североамериканской телекоммуникационной компании и были определены основные функции, которые должна выполнять разрабатываемая система.

При разработке базы данных CRM-системы для североамериканской телекоммуникационной компании были разработаны ER модель данных, логическая модель данных и в завершении проектирования базы данных была построена физическая модель данных. Для лучшего понимания взаимодействия разработанных модулей с базой данных была построена диаграмма последовательности.

После проектирования базы данных, необходимо было выбрать средство для непосредственной разработки CRM-системы для североамериканской телекоммуникационной компании. По результатам анализа и сравнения средств разработки, в качестве средства разработки был выбран язык программирования Java.

После выбора средств разработки было произведено описание основных принципы работы CRM-системы для североамериканской телекоммуникационной компании. Разработанная система позволит онлайн

подачу заявлений клиентов, а также обрабатывать их заявки и выводит данные о статусе их заявок в режиме онлайн, что позволит в большей степени снизить нагрузку на менеджера по продажам.

Далее было произведено тестирование CRM-системы для североамериканской телекоммуникационной компании методом «чёрного ящика», по результатам которого разработанная система была признана исправной и работоспособной. Пройдя тестирование методом «чёрного ящика» была проведена проверка на пригодность системы для использования на предприятии, по результатам данной проверки система была пригодна для использования.

Реализованная CRM-система для североамериканской телекоммуникационной компании имеет широкий функционал для управления клиентскими отношениями, тем самым обеспечивая сокращения временных затрат на взаимодействие с клиентами, что позволяет менеджеру по продажам сосредоточиться на более важных бизнес процессах.

Дальнейшее развитие CRM-системы для североамериканской телекоммуникационной компании предусматривает увеличение функциональных возможностей, например, отправка смс оповещений; сбор статистики и анализ данных о заявках клиента. А также модернизация CRM-системы для североамериканской телекоммуникационной компании по мере востребованности расширения управления клиентскими отношениями.

Список используемой литературы

Нормативно-правовые акты

1. ГОСТ 19.701-90. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения (ИСО 5807-85). Введ. 1992-01-01.- М.: Изд-во стандартов, 1992. – 14с.
2. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Термины и определения. Взамен ГОСТ 24.003-84, ГОСТ 22487-77; Введ. 1992-01-01.- М.: Изд-во стандартов, 1992. – 14с.
3. ГОСТ 34.320-96. Информационные технологии. Система стандартов по базам данных. Концепции и терминология для концептуальной схемы и информационной базы. Введ. 2001-07-01.- М.: Изд-во стандартов, 2001. – 46с. - (Основополагающие стандарты).
4. ГОСТ 34.602-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы. – Введ. 1990-01-01.-М.: Изд-во стандартов, 1990. – 12с. - (Основополагающие стандарты).

Научная и методическая литература

5. Вигерс, К. Разработка требований к программному обеспечению. 3е изд., дополнительное / К. Вигерс, Д. Битти., Пер. с англ. – М.: Издательство «Русская редакция»; СПб.: БХВ-Петербург, 2014. – 736 стр.
6. Голицина, О.Л., Максимов Н.В., Попов И.И. Базы данных: Учебное пособие. – М.: Формум: ИНФРА-М, 2013. – 352 с.
7. Гущина, О.М. Методические рекомендации к выполнению выпускной квалификационной работе бакалавра: учеб.-метод. пособие / О.М. Гущина, С.В. Мкртычев, А.В. Очеповский. – Тольятти : ТГУ, 2018. – 77 с.
8. Заботина, Н. Н. Проектирование информационных систем - М.: ДРОФА, 2013. - 336 с.

9. Колесов, Ю. Б. Моделирование систем. Объектно-ориентированный подход: учебное пособие / Ю. Б. Колесов, Ю. Б. Сениченков. – СПб.: БХВПетербург, 2012. – 192 с.
10. Кудинов, А. CRM. Практика эффективного бизнеса/ Кудинов А., Сорокин М., Голышева Е. – М.: 1С-Паблишинг, 2012. – 463с.
11. Леонтьев, Б.К. Web-дизайн: тонкости, хитрости, секреты / Б.К. Леонтьев.– М.: Майор, 2013. – 176с.
12. Мкртычев, С. В. Информационные системы в социальном менеджменте: учеб.пособие / С. В. Мкртычев ; ТГУ ; Ин-т математики, физики и информационных технологий ; каф. «Информатика и вычислительная техника». – ТГУ. – Тольятти : ТГУ, 2012. – 78 с.
13. Тарасов С. В. СУБД для программиста. Базы данных изнутри. —М.: СОЛОН-Пресс, 2015. — 320 с.: ил.
14. Никсон, Р. Создаем динамические веб-сайты с помощью Java, MySQL, JavaScript и CSS = Learning Java, MySQL, JavaScript and CSS / Р. Никсон ; [пер. с англ. Н. Вильчинский]. – 2-е изд. – Санкт-Петербург : Питер, 2013. – 560 с.

Электронные источники

15. Сайт проекта IDEF.RU [Электронный ресурс]: статья / «Проект компании «IDEF.RU»». Режим доступа: <http://www.ideal.ru/ideal.php>, свободный (дата обращения 26.01.2018). [http](http://www.ideal.ru/ideal.php)
16. Сайт проекта «Корпоративные хранилища данных. Интеграция систем. Проектная документация» [Электронный ресурс]: статья / «Типы моделей данных корпоративного хранилища данных», 2011. Режим доступа [http -exp.ru/dwh/dwh_model_types.php](http://www.ideal.ru/dwh/dwh_model_types.php) свободный (дата обращения 22.04.2018).
17. Сайт проекта «Национальный торговый университет» [Электронный ресурс]: статья / «Этапы проектирования ИС с применением UML». Режим доступа [http](http://www.ideal.ru/dwh/dwh_model_types.php) свободный (дата обращения 15.03.2018).

Литература на иностранном языке

18. Dag Øivind M., Daniel J.: Examining customer relationship management from a management fashion perspective, 2016.
19. Zahedan B.: Evaluation of Use of ERP in E-commerce: Methods and Strategies, 2014.
20. Parminder K. E-Marketing- A Global Perspective, 2015.
21. Bauer C. Java Persistence with Hibernate, 2nd Edition/ C. Bauer, G. Gregory, G. King, 2015
22. Cederholm D. CSS3 for Web Designers – A Book Apart 2014. – 142 с.
Gilmore W.J. Beginnig Java and MySQL. -3th Edition, 2012.

Приложение А

Фрагмент кода

```
<!-- taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" -->
<script src="{pageContext.request.contextPath}/resources/js/component/login.js" type="text/javascript"></script>
<div class="container">
  <div class="content">
    <font color="red">
      ${SPRING_SECURITY_LAST_EXCEPTION.message}
    </font>
    <form class="form-horizontal loginDiv" action="{request.getContextPath()}/appLogin" method="POST">
      <div class="form-group">
        <c:if test="{!empty errorMessageLogin}"><div class="alert alert-warning" role="alert">${errorMessageLogin}</div></c:if>
        <c:if test="{!empty errorMessagePassword}"><div class="alert alert-warning" role="alert">${errorMessagePassword}</div></c:if>
        <c:if test="{!empty errorMessageRegistration}"><div class="alert alert-warning" role="alert">${errorMessageRegistration}</div></c:if>
        <c:if test="{!empty successMessage}"><div class="alert alert-success" role="alert">${successMessage}</div></c:if>
        <label for="inputLogin">Login</label>
        <input class="form-control" type="text" name="app_username" id="inputLogin" class="inputLogin" placeholder="Login">
      </div>
      <div class="form-group">
        <label for="inputPassword">Password</label>
        <input class="form-control" type="password" name="app_password" id="inputPassword" class="inputPassword" placeholder="Password">
      </div>
      <div class="form-group">
        <div class="switch">
          Haven't account yet? Please, <a class="registerLink" href="#">register</a>
          <button type="submit" class="btn signIn">Sign in</button>
        </div>
      </div>
    </form>

    <form class="form-horizontal registerDiv" action="{request.getContextPath()}/register" method="POST" disabled hidden>
      <div class="form-group">
        <label for="inputRegisterLogin">Login</label>
        <div class="alert alert-warning jsErrorRegistrationLogin" role="alert" hidden>
          Login field must not be empty
        </div>
        <input class="form-control" type="text" name="registerLogin" id="inputRegisterLogin" class="jsInputRegisterLogin" placeholder="Login">
      </div>
      <div class="form-group">
        <label for="inputRegisterPassword">Password</label>
        <div class="alert alert-warning jsErrorRegistrationPassword" role="alert" hidden>
          Password field must not be empty
        </div>
        <div class="alert alert-warning jsErrorRegistrationPasswordNotMatch" role="alert" hidden>
          Password field doesn't match with Confirm password field
        </div>
        <input class="form-control" type="password" name="registerPassword" id="inputRegisterPassword" class="jsInputRegisterPassword" placeholder="Password">
      </div>
      <div class="form-group">
        <label for="inputRegisterPasswordAgain">Confirm your password</label>
        <div class="alert alert-warning jsErrorRegistrationPasswordAgain" role="alert" hidden>
          Confirm password field must not be empty
        </div>
        <input class="form-control" type="password" id="inputRegisterPasswordAgain" class="jsInputRegisterPasswordAgain" placeholder="Enter password again">
      </div>
      <div class="form-group">
        <div class="switch">

```

div.container > div.content > form.form-horizontal.loginDiv

Приложение Б

Фрагмент кода

```
1 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
2 <div class="row">
3   <c:forEach items="${services}" var="service">
4     <div class="col-sm-6 col-md-4">
5       <div class="thumbnail">
6         <div class="caption">
7           <form class="service-form" action="${request.getContextPath()}</buyService" method="post">
8             <input name="serviceId" type="hidden" class="form-control" value="${c:out value="${service.serviceId}"/>">
9             <h2>${service.name}</h2>
10            <input name="serviceName" type="hidden" class="form-control" value="${c:out value="${service.name}"/>">
11            <input name="serviceType" type="hidden" class="form-control" value="${c:out value="${service.type}"/>">
12            <h3>${service.price}$</month</h3>
13            <input name="servicePrice" type="hidden" class="form-control" value="${c:out value="${service.price}"/>">
14            <p>${service.description}</p>
15            <input name="serviceDescription" type="hidden" class="form-control" value="${c:out value="${service.description}"/>">
16            <p><button type="submit" class="btn btn-default">Buy</button></p>
17          </form>
18        </div>
19      </div>
20    </div>
21  </c:forEach>
22 </div>
23
```

```
package com.concretepage.controller;

import ...

@Controller
public class ServiceController {
    private static final String PENDING = "pending";
    private static final String SUCCESS_MESSAGE = "You have successfully applied the request. Manager will contact you by phone for further information";
    @Autowired
    private ServiceDAOService service;
    @Autowired
    private IUserService userService;
    @Autowired
    private RequestService requestService;

    @RequestMapping(value = "/internet")
    @ResponseBody
    public ModelAndView getInternetServices(ModelMap model) {
        model.addAttribute( attributeName: "services", service.getServicesByType("internet"));
        return new ModelAndView( viewName: "/view/component/services" );
    }

    @RequestMapping(value = "/tv")
    @ResponseBody
    public ModelAndView getTvServices(ModelMap model) {
        model.addAttribute( attributeName: "services", service.getServicesByType("tv"));
        return new ModelAndView( viewName: "/view/component/services" );
    }

    @RequestMapping(value = "/phone")
    @ResponseBody
    public ModelAndView getPhoneServices(ModelMap model) {
        model.addAttribute( attributeName: "services", service.getServicesByType("phone"));
        return new ModelAndView( viewName: "/view/component/services" );
    }
}
```

Приложение В

Фрагмент кода

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<div class="container buy-container">
  <form class="form-horizontal buy-form" action="<%=request.getContextPath()%>/addRequest" method="POST">
    <div class="form-group">
      <label class="col-sm-2 control-label">Service</label>
      <div class="col-sm-6">
        <label class="control-label">${service.type} ${service.name}</label>
      </div>
      <input name="serviceId" type="hidden" class="form-control" value="<c:out value="${service.serviceId}"/>">
    </div>
    <div class="form-group">
      <label class="col-sm-2 control-label">Price/Month</label>
      <div class="col-sm-6">
        <label class="control-label">${service.price}$</label>
      </div>
    </div>
    <div class="form-group">
      <label class="col-sm-2 control-label">Description</label>
      <div class="col-sm-6">
        <label class="control-label">${service.description}</label>
      </div>
    </div>
    <div class="form-group">
      <label class="col-sm-2 control-label">Your address</label>
      <div class="col-sm-6">
        <label class="control-label">
          <c:choose>
            <c:when test="&${not empty user.address}">
              <c:out value="&${user.address}"/>
            </c:when>
            <c:otherwise>
              You can't apply the request until you add address in profile
            </c:otherwise>
          </c:choose>
        </label>
      </div>
    </div>
    <div class="form-group">
      <label class="col-sm-2 control-label">Your phone number</label>
      <div class="col-sm-6">
        <label class="control-label">
          <c:choose>
            <c:when test="&${not empty user.phoneNumber}">
              <c:out value="&${user.phoneNumber}"/>
            </c:when>
            <c:otherwise>
              You can't apply the request until you add phone number in profile
            </c:otherwise>
          </c:choose>
        </label>
      </div>
    </div>
    <div class="form-group">
      <input type="submit" value="Add Request" class="btn btn-primary">
    </div>
  </form>
</div>
div.container.buy-container
```

Приложение Г

Фрагмент кода

```
<%@ taglib uri = "http://java.sun.com/jsp/jstl/core" prefix = "c" %>
<h2 class="sub-header">Your requests</h2>
<div class="table-responsive">
  <table class="table table-striped">
    <thead>
      <tr>
        <th>Service</th>
        <th>Type</th>
        <th>Price/Month</th>
        <th>Status</th>
      </tr>
    </thead>
    <c:forEach items="${requests}" var="request">
      <tr>
        <td>${request.serviceName}</td>
        <td>${request.serviceType}</td>
        <td>${request.servicePrice}$</td>
        <td>${request.requestStatus}</td>
      </tr>
    </c:forEach>
  </tbody>
</table>
</div>
```

```
package com.concretepage.controller;

import ...

@Controller
public class RequestController {
    private static final String ROLE_USER = "ROLE_USER";
    private static final String ROLE_ADMIN = "ROLE_ADMIN";

    @Autowired
    private RequestService requestService;
    @Autowired
    private IUserService userService;

    @RequestMapping(value = "/changeStatus", method = RequestMethod.POST)
    @ResponseBody
    public void changeRequestStatus(@RequestParam("requestStatus") String requestStatus, @RequestParam("requestId") Integer requestId) {
        requestService.updateStatusRequest(requestId, requestStatus);
    }

    @RequestMapping(value = "/requests")
    @ResponseBody
    public ModelAndView showRequestsAjax(ModelMap model, Authentication authentication) {
        List<RequestDTO> requests;
        if (authentication != null) {
            UserInfo user = userService.getDataByUsername(authentication.getName());
            if (user.getRole().equals(ROLE_USER)) {
                Integer userId = user.getUserId();
                requests = requestService.getRequestsDTOForUser(userId);
                model.addAttribute( attributeName: "requests", requests);
                return new ModelAndView( viewNames: "/view/component/userRequests" );
            } else if (user.getRole().equals(ROLE_ADMIN)) {
                requests = requestService.getRequestsDTOForAdmin();
                model.addAttribute( attributeName: "requests", requests);
                return new ModelAndView( viewNames: "/view/component/adminRequests" );
            }
        }
        return new ModelAndView( viewNames: "/view/page/access-denied");
    }
}
```