

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

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

Кафедра Прикладная математика и информатика
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки, специальности)

Бизнес-информатика
(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка проекта внедрения CRM-системы предприятия»

Обучающийся

И.Г. Тараник

(Инициалы Фамилия)

(личная подпись)

Руководитель

Н.Н. Рогова

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2022

Аннотация

Ключевые слова: CRM-система, автоматизация, бизнес-процессы, банковское обслуживание, веб-технологии, php, mysql, javascript.

Общие сведения о работе: 56 страниц, 34 рисунка, 8 таблиц, 5 приложений, 21 библиографический источник.

Объектом исследования являются бизнес-процессы отдела дистанционного обслуживания (ДБО) ПАО Банк «Александровский» и их автоматизация посредством разработки и внедрения CRM-системы.

Цель выпускной квалификационной работы – разработка CRM-системы отдела ДБО ПАО Банк «Александровский» с использованием веб-технологий.

При анализе бизнес-процессов отдела ДБО были выявлены методики работы и аппаратное обеспечение отдела, сделаны выводы об эффективности применения существующих средств и инструментов. В ходе анализа были выявлены слабые стороны в существующей системе и предложены пути улучшения. В результате чего были сформулированы функциональные требования к разрабатываемой системе.

При сравнительном анализе существующих решений на рынке CRM систем были сделаны выводы о целесообразности разработки собственной системы.

Новизной и оригинальностью работы является разработка CRM-системы на базе веб-технологий, что позволит использовать систему удалённо посредством сети интернет, что в значительной степени упростит эксплуатацию данной системы и увеличит удобство пользования.

По результатам внедрения разработанной CRM была достигнута оптимизация работы отдела ДБО, повышение конверсий от продажи услуг, учёт деятельности и оценка продуктивности работы специалистов отдела.

Оглавление

Глава 1 Теоретическая часть.....	6
1.1 Анализ предметной области ПАО Банк «Александровский».....	6
1.2 Анализ бизнес-процессов отдела ДБО и обоснование необходимости разработки CRM системы	10
1.3 Обзор существующих CRM систем и постановка задачи.....	15
Глава 2 Проектная часть	22
2.1 Описание информационных потоков и событий системы	22
2.2 Выбор технологий и инструментов реализации	25
2.3 Разработка базы данных.....	29
2.4 Разработка интерфейса и фронтенд-части	34
2.5 Разработка серверной части.....	35
Глава 3 Анализ работы CRM.....	39
3.1 Контрольный пример использования	39
3.2 Расчёт экономической эффективности.....	45
Заключение	53
Список используемой литературы	55
Приложение А Дамп базы данных	57
Приложение Б Листинг функции dbGo()	61
Приложение В Листинг функции sql().....	62
Приложение Г Листинг функции sqlInset().....	63
Приложение Д Листинг функции template().....	64

Введение

С точки зрения управления бизнес-процессами, очень важным моментом является строгий учёт и систематизация различных информационных потоков, связанных с его деятельностью. Современные компании для этих целей используют различные программные средства, которые представлены на рынке, или же разрабатывают свои. Одним из примеров программных средств, решающих задачи бизнеса в области управления и учёта данных, являются системы CRM (Customer Relationship Management) – системы для управления взаимоотношениями с клиентами.

Внедрение CRM-системы является одним из приоритетных направлений развития любой компании, поскольку сможет обеспечить повышение качества обслуживания клиентов, уменьшить трудозатраты на сопровождение и освободить сотрудников от рутинной работы.

Веб-приложения, как правило, имеют клиент-серверную архитектуру, как наиболее популярную и востребованную.

Исходя из актуальности применения веб-технологий в построении CRM-систем, в данной работе будет спроектирована и построена CRM-система на базе веб-технологий.

Актуальность данной выпускной квалификационной работы заключается в том, что внедрение системы управления взаимоотношениями с клиентами прежде всего повысит эффективность работы отдела ДБО ПАО Банк «Александровский».

Целью выпускной квалификационной работы является разработка CRM-системы для работы отдела ДБО ПАО Банк «Александровский».

Объектом исследования является отдел ДБО ПАО Банк «Александровский», который предоставляет клиентам различные решения в области дистанционного банковского обслуживания.

Предметом исследования выступает система CRM, автоматизирующая бизнес-процессы внутри отдела.

Для достижения поставленной цели необходимо исследовать существующую модель бизнес-процессов в отделе ДБО и выявить участки, требующие решений автоматизации, предложить способы решения выявленных проблем, провести сравнительный анализ существующих на рынке решений, выбрать лучшие практики применения существующих систем и использовать их в разработке, выбрать технологии для разработки собственной системы, разработать систему CRM, внедрить разработанную систему в отдел ДБО, и в дальнейшем оценить экономический эффект от разработки.

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

В первой главе описано изучение организационной структуры и анализ бизнес-процессов ПАО банк «Александровский» в целом и отдела ДБО в частности, поставлена задача на разработку собственной системы CRM.

Во второй главе описана проектная часть, где выбраны технологии для разработки и произведено проектирование системы.

В третьей главе приведено экономическое обоснование спроектированной системы, описана инструкция по запуску системы и описан контрольный пример использования системы на практике.

В заключении подведены итоги выполненной работы, описана практическая значимость разработанной CRM-системы.

Глава 1 Теоретическая часть

1.1 Анализ предметной области ПАО Банк «Александровский»

Предметом рассмотрения данной работы является ПАО Банк «Александровский» и в частности отдел дистанционного банковского обслуживания клиентов (ДБО). Банк и отдел рассматриваются с точки зрения ведения бизнес-процессов с целью выявления элементов, деятельность которых может быть улучшена методами автоматизации.

В своей деятельности ПАО Банк «Александровский» руководствуется уставом и нормативными документами:

- кодекс корпоративного управления от 04.05.2021;
- устав ПАО Банк «Александровский»;
- изменения, вносимые в устав №1, №2, №3.

Организационная структура банка представлена на рисунке 1.

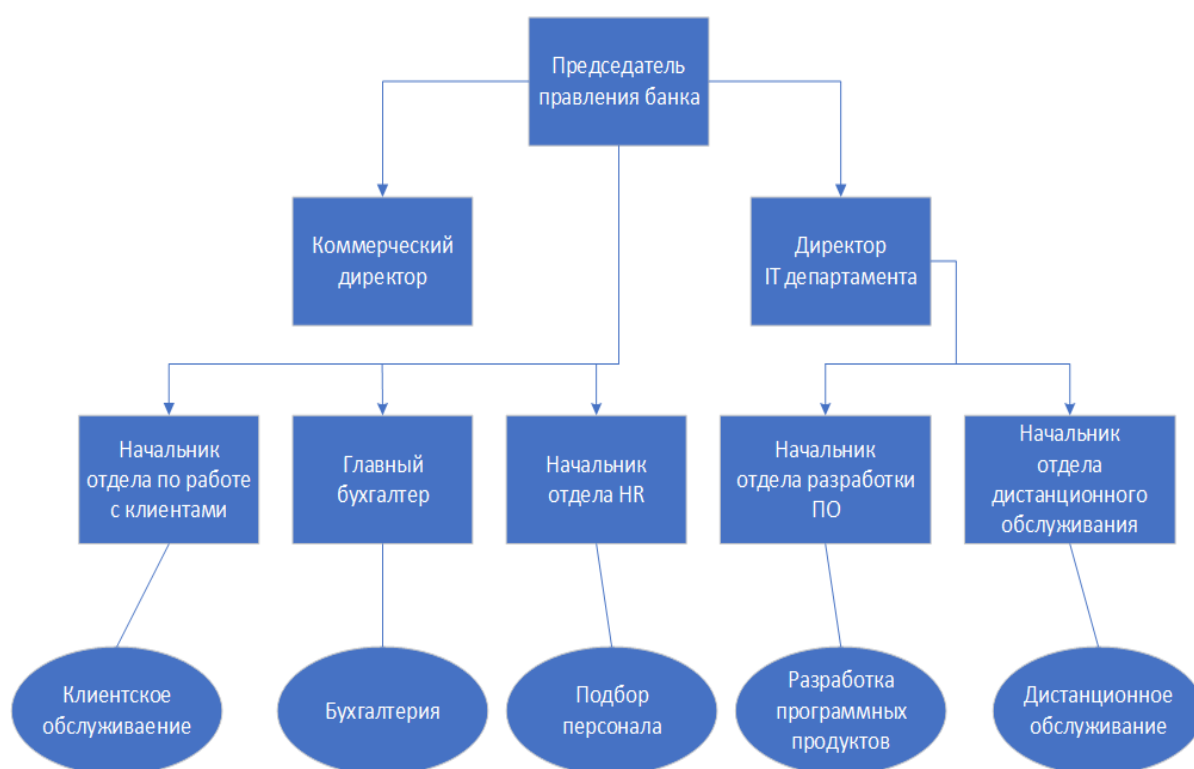


Рисунок 1 – Организационная структура органов управления

Председатель правления банка имеет высший приоритет в структуре управления. Ему подчиняются непосредственно все отделения банка, а также коммерческий директор и директор IT-департамента. Каждый из отделов банка подчиняется непосредственно председателю правления, а подразделения и сотрудники внутри отделений подчиняются непосредственно начальникам каждого из отделов.

Рассмотрим более подробно структуру отдела дистанционного банковского обслуживания, представленную на рисунке 2.



Рисунок 2 – Организационная структура отдела ДБО

Начальник отдела ДБО непосредственно осуществляет установку и обновление серверов банк-клиента, поддержание их в соответствующем действующему законодательству состоянии, исправление ошибок, которые можно починить без привлечения разработчиков, взаимодействие с разработчиками (заявки на доработку функционала, вопросы по взаимодействию банк-клиента и автоматизированной банковской системы), взаимодействие со смежными отделами в части реализации новых функций банк-клиента (в основном, с теми, которые занимаются автоматизированной

банковской системой). Начальнику отдела ДБО подчиняются: главный специалист, ведущий специалист, специалист.

Главный специалист осуществляет тестирование новой версии банк-клиента перед обновлением сервера, публикации приложений банка в AppStore и GooglePlay, написание пользовательских инструкций (для сотрудников и для клиентов), приём и обработку документов от операционного отдела и контакт-центра, взаимодействие с разработчиками (в основном, сообщения об ошибках и заявки на доработку функционала), управление учётными записями сотрудников банка в системе банк-клиент, управление учётными записями клиентов банка, обучение сотрудников.

Ведущий специалист осуществляет приём и обработку документов от операционного отдела и контакт-центра, передачу информации об ошибках в системе банк-клиент разработчикам, управление учётными записями клиентов банка, консультирование клиентов и сотрудников по телефону.

Специалист осуществляет приём и обработку документов от операционного отдела и контакт-центра, управление учётными записями клиентов банка, консультирование клиентов и сотрудников по телефону.

ДБО – это возможность, которая предоставляется клиентам по банковскому обслуживанию, не приходя в банк, используя различные каналы связи. Клиенты банка могут управлять счетами, а также получать различные консультации.

Перечень услуг отдела дистанционного обслуживания представлен на рисунке 3.

Как видно из рисунка 3 – банк предоставляет не только услуги дистанционного обслуживания, но и обеспечивает клиентскую и техническую поддержку своих продуктов.

Современные методы ведения бизнес-процессов позволяют банку и, в частности, отделу ДБО успешно конкурировать на рынке дистанционного банковского обслуживания.

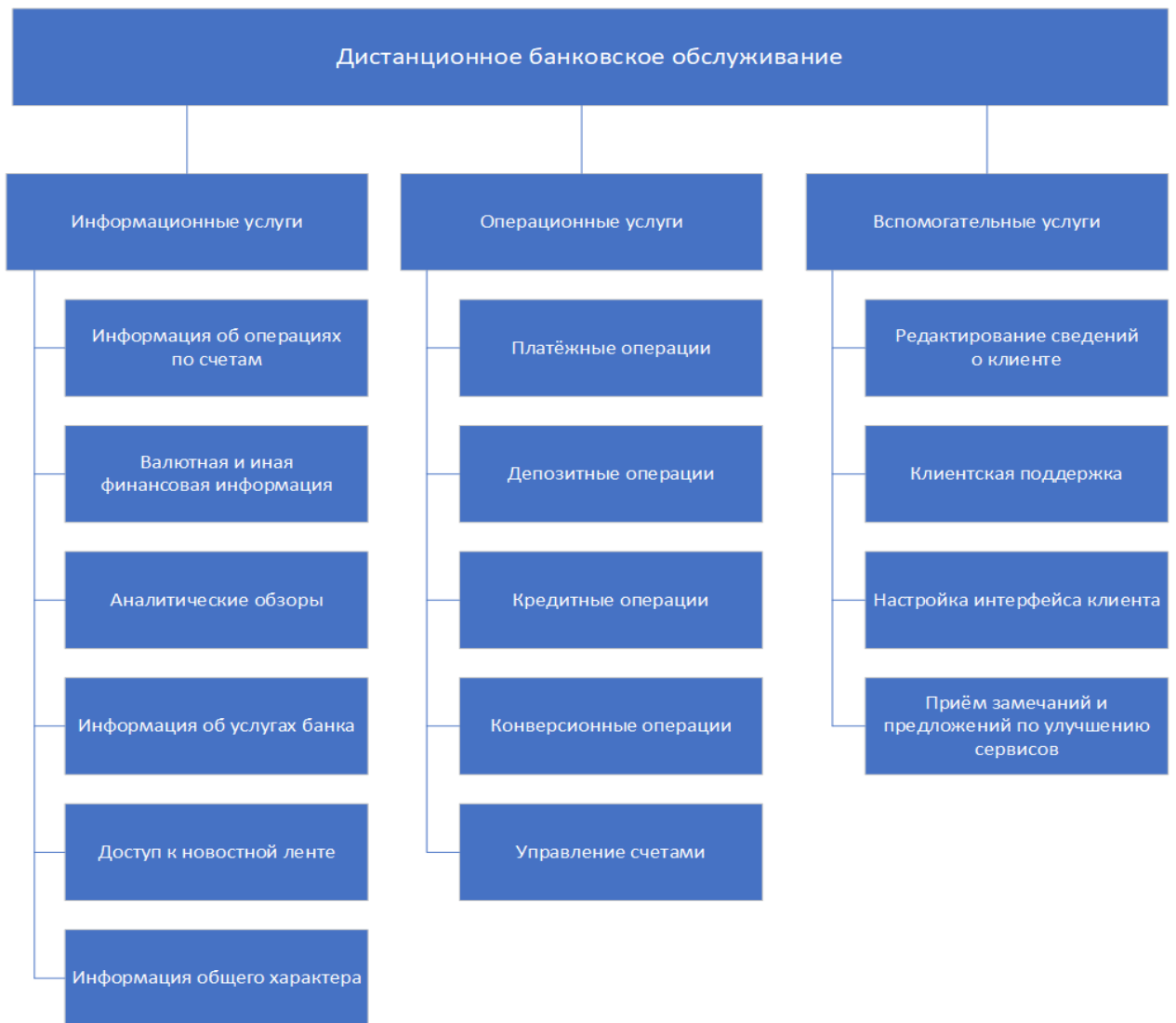


Рисунок 3 – Перечень услуг ДБО

В современном мире ДБО имеет три основные формы: системы телебанк, Клиент-Банк, интернет-банкинг.

ПАО Банк «Александровский» входит в десятку лидеров России в оказании услуг ДБО.

Данный вид обслуживания имеет ряд преимуществ:

- увеличение количества клиентов,
- сокращение операционных издержек,
- экономия времени,
- комфорт при взаимодействии клиента с банком,

- увеличение популярности банка.

Специалист дистанционного банковского обслуживания выполняет такие обязанности:

- ввод данных о клиенте в систему ДБО,
- проверка соответствия подписи во всех документах,
- сверка данных о клиенте,
- проверка документов клиента,
- проверка правильности оформления документов,
- консультирование клиентов,
- актуализация документации по системе ДБО.

1.2 Анализ бизнес-процессов отдела ДБО и обоснование необходимости разработки CRM системы

Как видно из описания системы ДБО в данном банке, она подразумевает использование программных средств и технологий как со стороны банка, так и со стороны клиента. Рассмотрим организацию рабочих мест с точки зрения аппаратных и программных средств, используемых в отделе ДБО.

Работа сотрудников отдела организована посредством персональных компьютеров, объединённых в локальную сеть. Характеристики компьютеров сети, следующие:

- операционная система семейства Windows 10 или выше,
- последние установленные обновления для корректной работы,
- компьютер с процессором от 1.5 ГГц,
- оперативная память не менее 1 Гб,
- монитор,
- принтер,
- установленный современный веб-браузер,
- доступ в интернет,

– локальный веб-сервер.

Локальный веб сервер хранит в себе набор документов – результаты работы сотрудников отделения. Персонал осуществляет фиксацию каждого действия, заполняя таблицы Microsoft Excel и формируя документы Microsoft Word. (Рисунок 4). Таким образом, анализ действий сотрудников, сами действия сотрудников и прочие бизнес-процессы внутри отдела целиком полагаются на правильность ведения документооборота сотрудниками, то есть, подвержены человеческому фактору. Результаты общения с клиентами и показатели различных конверсий контролируются ведущим специалистом отдела.

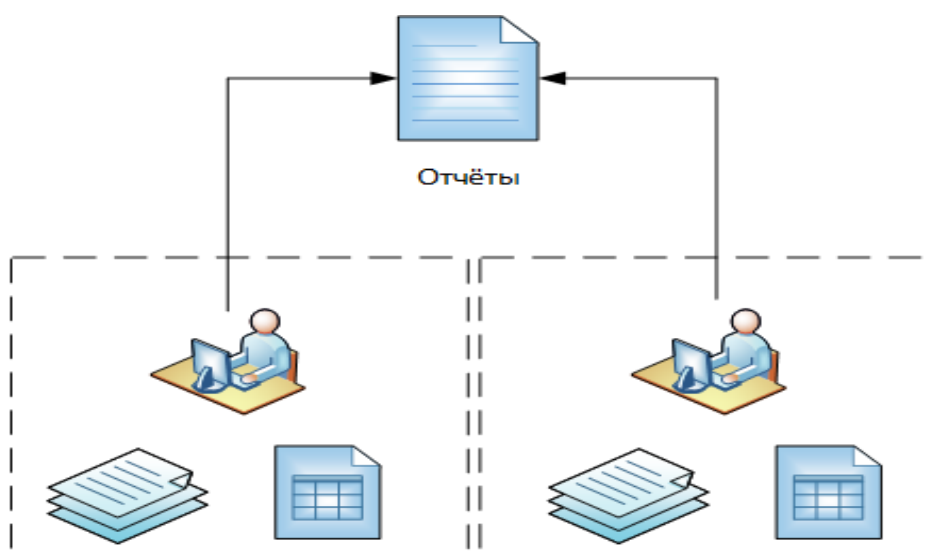


Рисунок 4 – Организация рабочих мест в отделе ДБО

Для анализа деятельности отдела ДБО с точки зрения бизнес-процессов применим стандарт IDEF0. Стандарт IDEF0 рассчитан для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающие эти функции [2]. Применяя данный стандарт и учитывая анализ деятельности отдела ДБО получим контекстную диаграмму (рисунок 5).

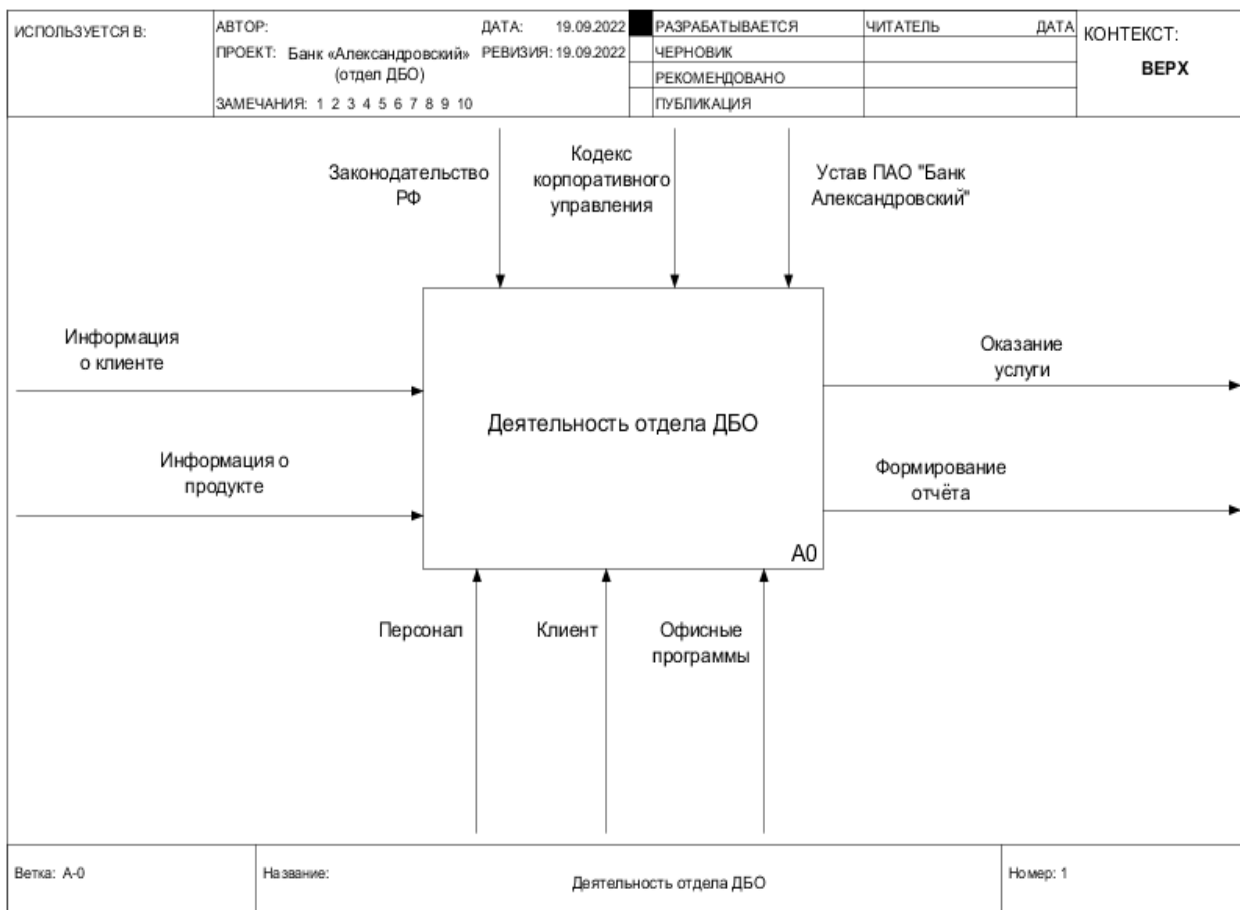


Рисунок 5 – Контекстная диаграмма деятельности отдела ДБО («как есть»)

На контекстной диаграмме представлены основные направления деятельности отдела ДБО, а именно – оказание услуг дистанционного банковского обслуживания и формирование отчётов по результатам деятельности для внутреннего пользования. Необходимо провести декомпозицию верхнего уровня контекстной диаграммы для детализации процессов. Полученная декомпозиция представлена на рисунке 6.

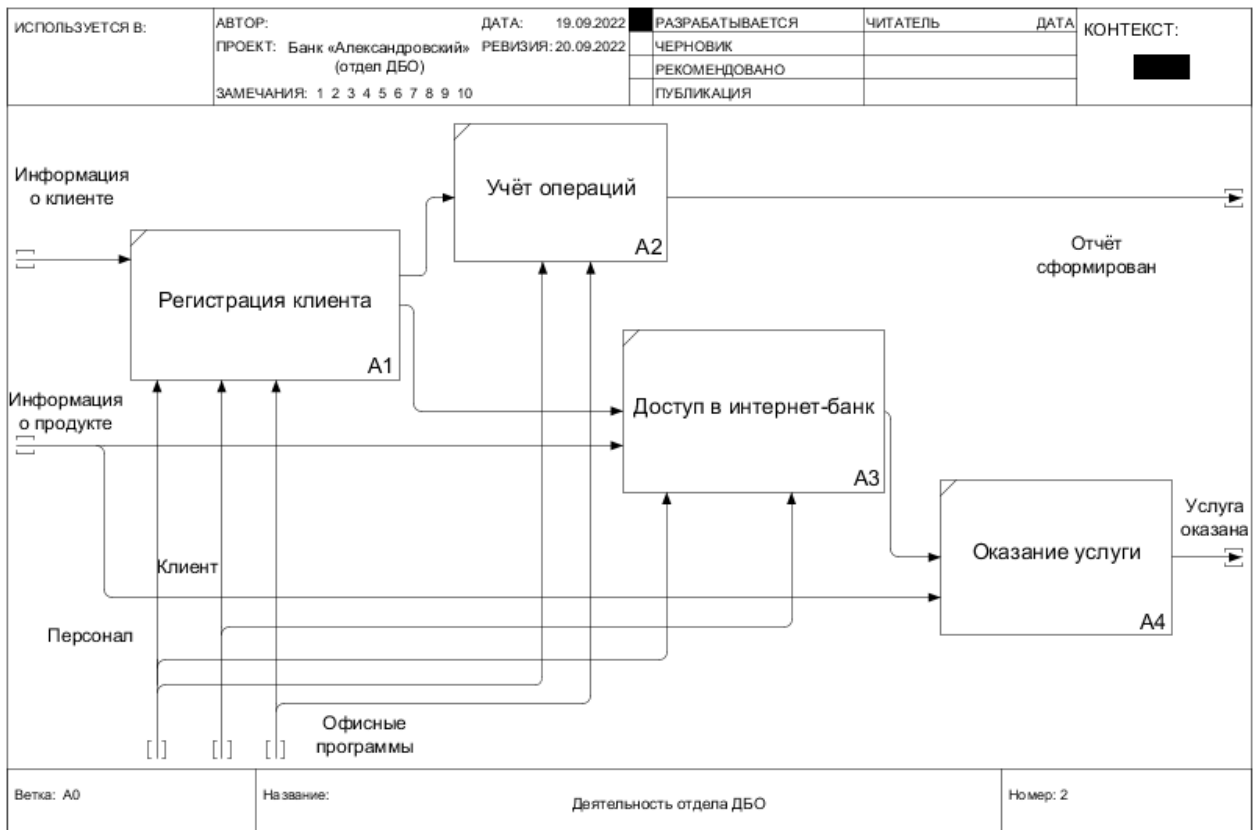


Рисунок 6 – Декомпозиция верхнего уровня деятельности отдела ДБО

Из полученной декомпозиции верхнего уровня видно, что участие менеджеров необходимо на каждом из этапов процесса деятельности, в том числе и при выполнении операций учёта и оформления различных заявок. Таким образом, система взаимоотношений с клиентами с точки зрения автоматизации выстроена довольно слабым образом, без применения специальных программных средств, а именно: учёт заявок клиентов, оценка эффективности работы специалистов, входящие и исходящие звонки, и прочие вопросы метрик взаимоотношений с клиентами решаются специалистами в индивидуальном порядке, не используя системный подход.

Всё рассмотренное выше позволяет рассуждать о том, что отделу ДБО требуется специализированное решение в области взаимоотношений с клиентами и ведения бизнес-процессов, а именно CRM-система.

На основании выявленных недостатков существующей системы можно смоделировать контекстную диаграмму деятельности отдела ДБО по типу «как должно быть» (рисунок 7). Из данной диаграммы можно увидеть, что часть управляющих и документирующих функций берёт на себя система CRM.

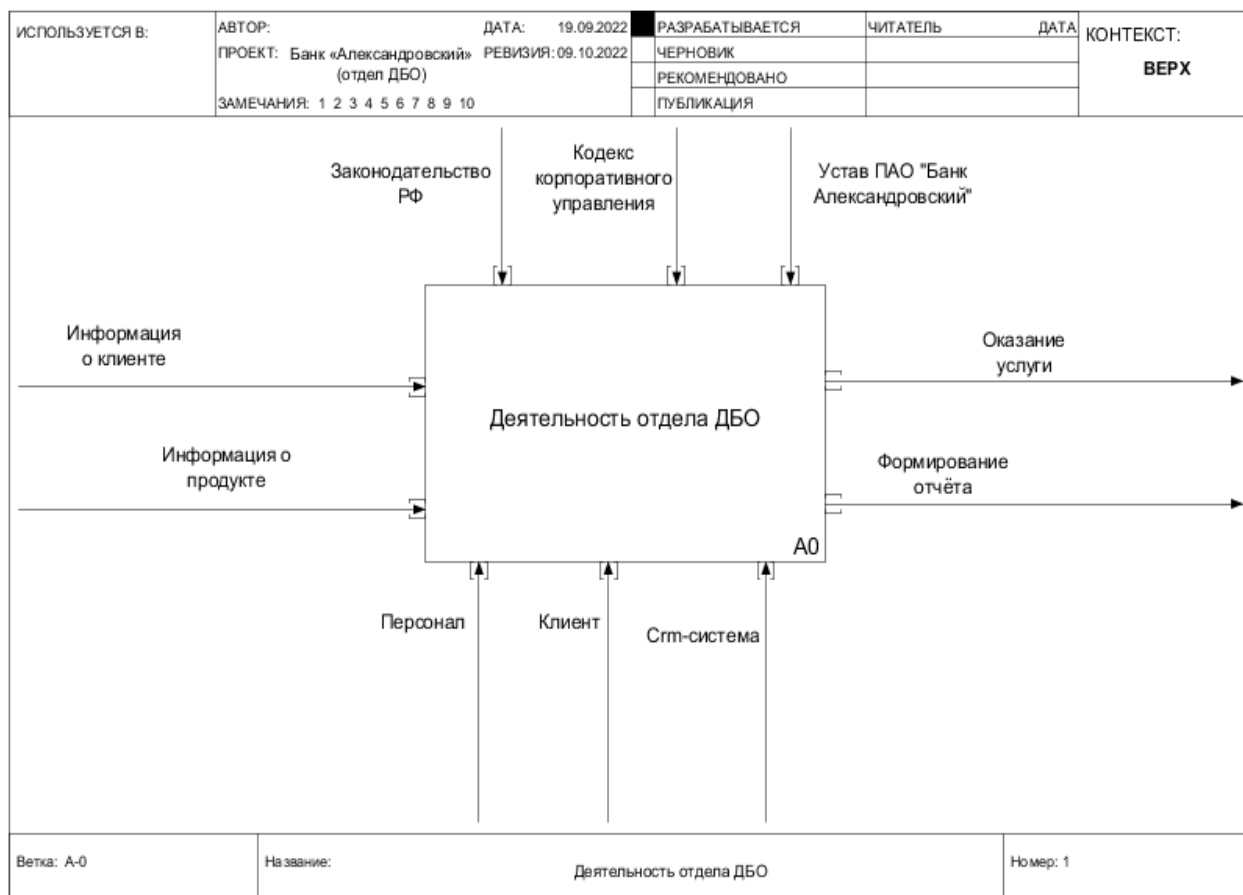


Рисунок 7 – Контекстная диаграмма деятельности отдела ДБО («как должно быть»)

Как видно из рисунка 7 – использование офисных программ в деятельности отдела ДБО должно быть сокращено в пользу использования системы CRM.

Также для наглядности рассмотрим диаграмму декомпозиции верхнего уровня «как должно быть». Данная диаграмма представлена на рисунке 8.

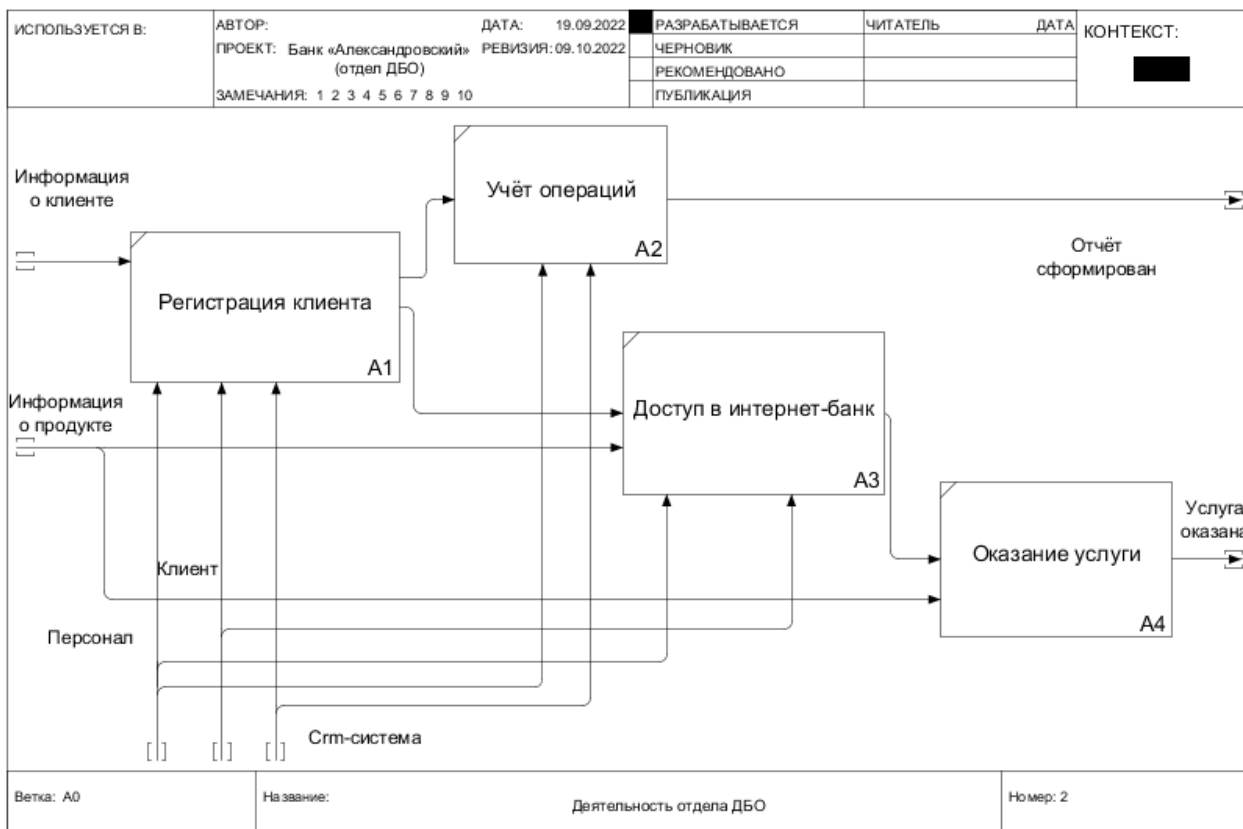


Рисунок 8 – Декомпозиция верхнего уровня деятельности отдела ДБО («как должно быть»)

Диаграмма на рисунке 8 раскрывает детали ведения деятельности отдела ДБО и показывает отсутствие средств автоматизации на отдельных этапах процесса предоставления услуги.

1.3 Обзор существующих CRM систем и постановка задачи

Обзор существующих решений CRM позволит выявить сильные и слабые места в предлагаемом функционале и выявить необходимость применения того или иного решения в отделе ДБО, либо же будет предложена разработка собственной системы.

Среди крупнейших мировых производителей CRM на российском рынке представлены Oracle, SAP, Siebel Systems, Microsoft, SalesLogix, Terrasoft. Набирают популярность и отечественные продукты, пока востребованные в

основном представителями среднего и малого бизнеса [10]. Среди них, например, Bitrix24, AmoCRM, «Мегаплан» и ряд других.

Построение систем для крупных компаний требует огромных ресурсов со стороны разработчиков и огромное количество человеко/часов на разработку, что не может быть рассмотрено в рамках данной работы. В рамках данной работы будут рассмотрены решения для малого и среднего бизнеса. Рассмотрим наиболее популярные в этом сегменте системы – Bitrix24 и AmoCRM.

CRM-система Битрикс24 – это комплекс инструментов для нужд малого бизнеса. Для работы Битрикс24 достаточно браузера и доступа в интернет[2]. Внешний вид приложения Битрикс24 и основной функционал работы представлен на рисунках 9-11.

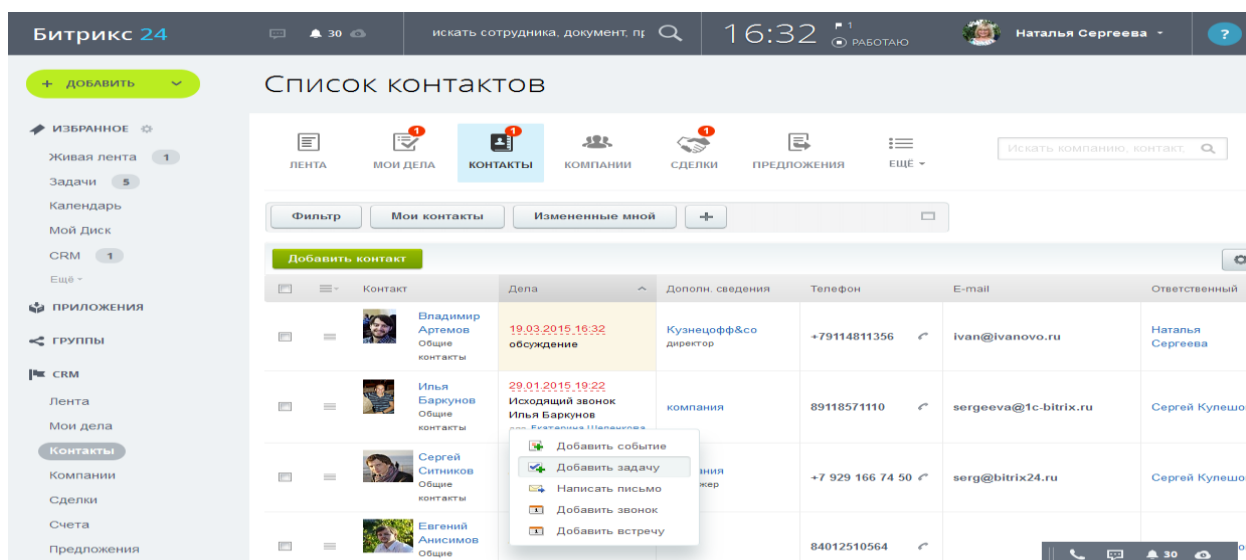


Рисунок 9 – Внешний вид CRM Битрикс-24

В Битрикс24 вся необходимая информация по клиентам, по сделкам, по проектам доступна из панели управления. В процессе пользования системой можно наблюдать за ходом и этапами продвижения сделки, планировать различные события и выстраивать их в календаре. Оформлять контакты можно загружая большое количество информации в профиль, аналогично профилю в

социальных сетях. Наблюдать за событиями и произведёнными действиями можно в специальной ленте.

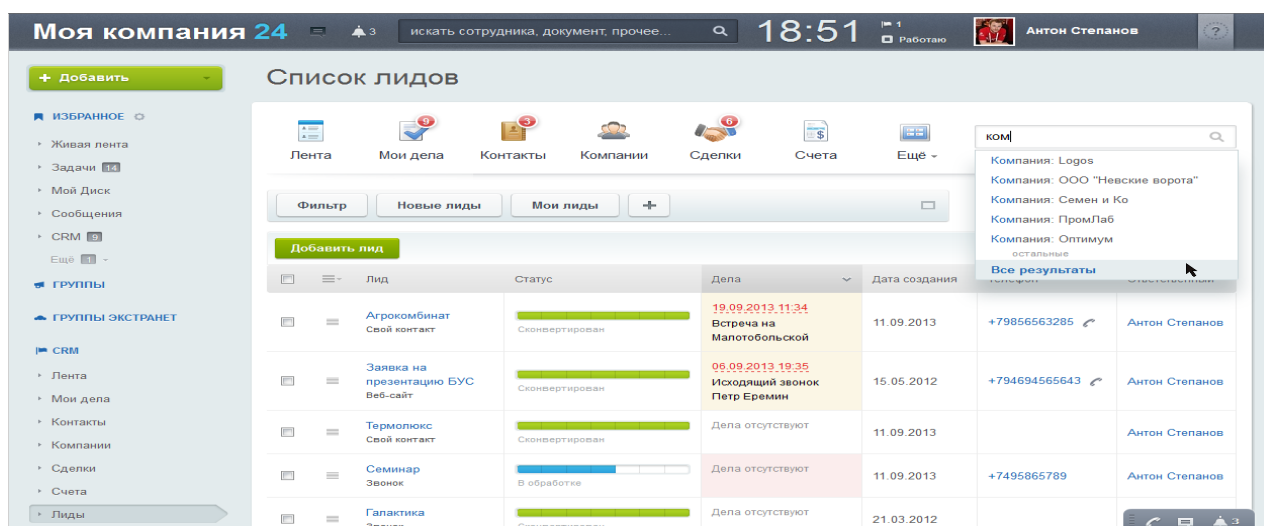


Рисунок 10 – Хранилище данных в CRM Битрикс-24

Битрикс24 систематизирует информацию по клиентам, задачам и проектам, являясь единым хранилищем данных. Каждый шаг по сделке и работе с клиентом фиксируется. Администратор может видеть, как работают менеджеры, как они ведут сделки, как общаются с клиентами по телефону.

В CRM Битрикс24 можно анализировать работу в целом по отделу, по конкретным клиентам и по отдельным менеджерам. В систему встроена IP-телефония. Звонки клиентам на мобильные или стационарные телефоны осуществляются сразу из CRM. Все входящие и исходящие телефонные разговоры записываются и сохраняются в карточке клиента. Таким образом можно проверить, как менеджеры общаются с клиентами, соблюдают ли они регламенты.

По пропущенным звонкам в CRM-системе автоматически создается задача менеджеру.

С точки зрения аналитики можно наблюдать воронку продаж за любой временной промежуток, в разрезе по сделкам, по отдельным менеджерам, по направлениям. Аналитический режим даст больше представления о том, как

идут дела компании и что срочно нужно скорректировать, какой менеджер отстаёт, а кого, наоборот, нужно мотивировать на ещё большие результаты.

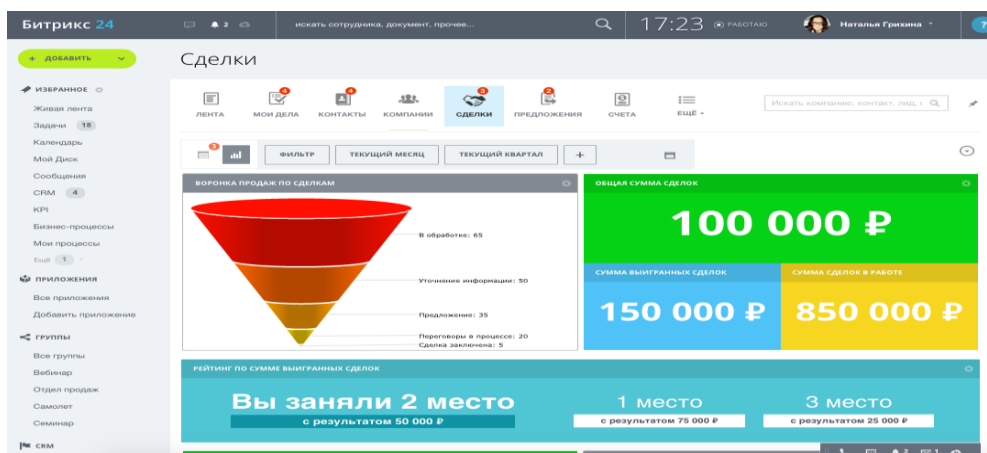


Рисунок 11 – Воронка продаж в CRM Битрикс-24

Как видно из описания – CRM Битрикс-24 является полноценной системой для использования в задачах малого бизнеса, что и обуславливает её популярность на рынке.

АmoCRM – это облачный сервис для автоматизации работы с клиентами, позиционирующий себя как «CRM для отделов продаж» [1]. Внешний вид приложения АmoCRM представлен на рисунках 10-12.

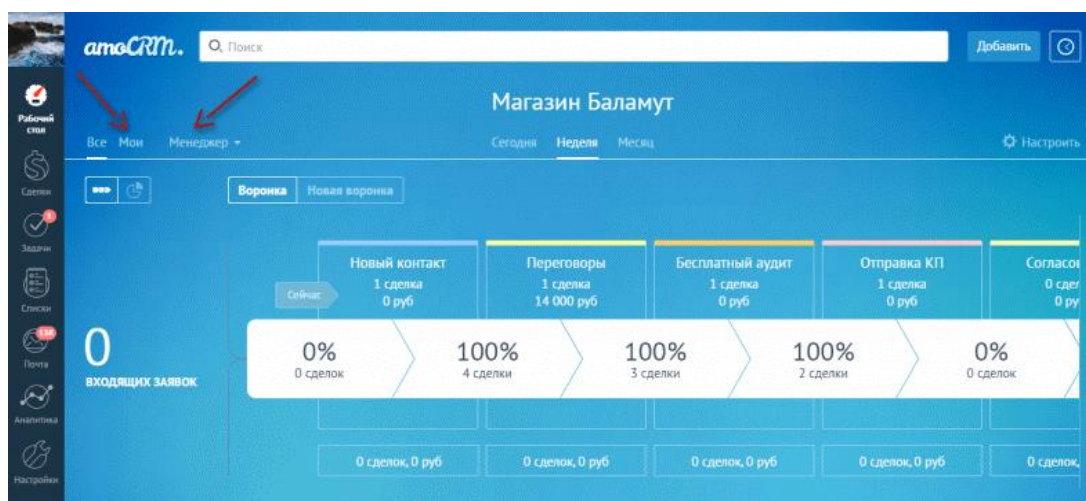


Рисунок 12 – Внешний вид АmoCRM

Одним из основных преимуществ AmoCRM является так называемая «воронка продаж» [16]. Это этапы прохождения сделки от момента регистрации до момента оформления. Воронку можно настроить под отдельные задачи бизнеса, создавать большое их количество и тонко настраивать. Преимущество воронки заключается в визуальной наглядности такого подхода и возможности автоматизировать большое количество задач.

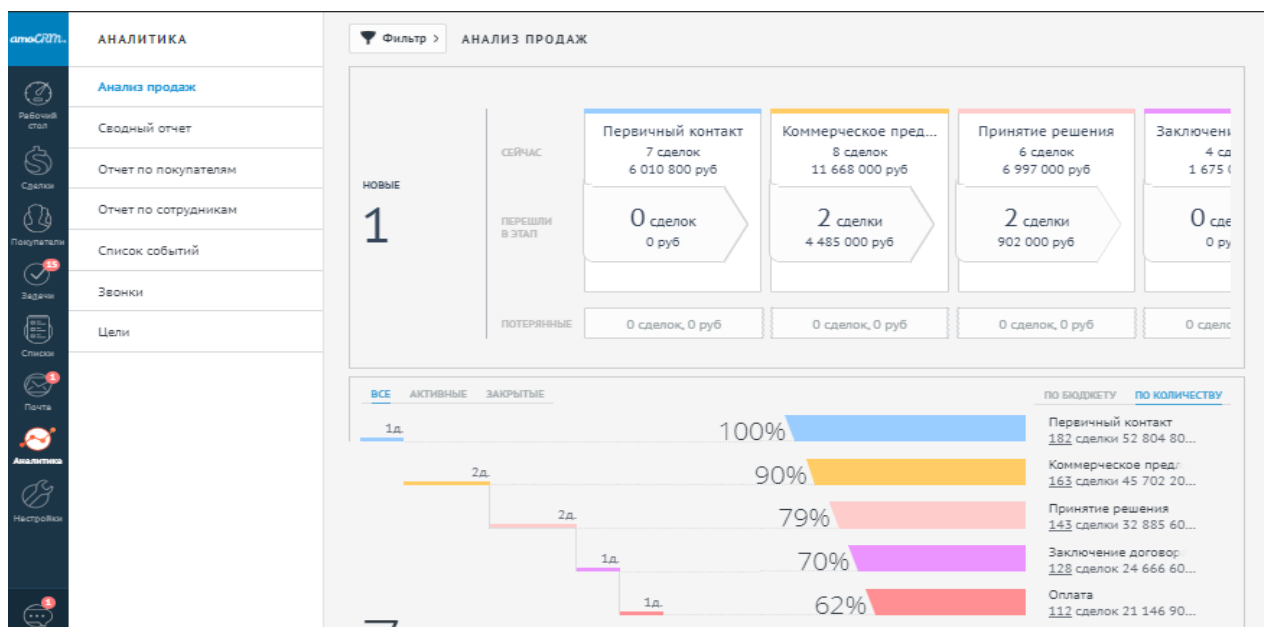


Рисунок 13 – Воронка продаж AmoCRM

Также в AmoCRM есть встроенная телефония с большим количеством операторов сотовой связи.

В AmoCRM существуют авторизованные сделки с возможностью управления приоритетами каждой из таких сделок. По достаточно большому количеству атрибутов можно производить фильтрацию и сортировку сделок. AmoCRM позволяет интегрировать множество сторонних сервисов и решений, таких как социальные сети, мессенджеры, сайты. В AmoCRM есть мобильная версия CRM-системы, в которой можно использовать функцию сканера визиток. Она нужна для того, чтобы сразу вносить данные о партнере

или клиенте в базу данных. Очень полезная функция для тех, кто налаживает деловые связи на мероприятиях или таким образом генерирует трафик.

Также доступна расширенная аналитика по различным критериям.

Screenshot of the AmoCRM analytics interface. The main content area displays a report titled "ОТЧЕТ ПО ВСЕМ СОТРУДНИКАМ ТЕКУЩИЙ ГОД 1 ЯНВ 2017 – 31 ДЕК". The report table has the following columns: НАЗВАНИЕ, СОЗДАНО СДЕЛОК, ВЫИГРАНО СДЕЛОК, АКТИВНЫХ СДЕЛОК, and ЗАДАЧ В РАБОТЕ. The data rows are as follows:

НАЗВАНИЕ	СОЗДАНО СДЕЛОК	ВЫИГРАНО СДЕЛОК	АКТИВНЫХ СДЕЛОК	ЗАДАЧ В РАБОТЕ
Екатерина Ищенко	3 (7 400 руб)		3 (7 400 руб)	
Мария Михалкова	41 (9 643 199 руб)	21 (5 820 000 руб)	41 (9 643 199 руб)	
Михаил Антипин	15 (5 911 000 руб)	4 (4 421 000 руб)	15 (5 911 000 руб)	
Яна Трофина	20 (9 532 200 руб)	2 (87 000 руб)	20 (9 532 200 руб)	
Александра Елисеев	12 (2 119 500 руб)	1 (4 500 руб)	12 (2 119 500 руб)	
Максим Кожухов	14 (5 147 000 руб)	3 (4 447 000 руб)	14 (5 147 000 руб)	
Елена Антонова	25 (20 541 030 руб)	7 (8 815 980 руб)	25 (20 541 030 руб)	
Денис Звягин	5 (4 420 000 руб)	2 (4 390 000 руб)	5 (4 420 000 руб)	
Татьяна Сергеева	16 (11 784 000 руб)	1 (48 000 руб)	16 (11 784 000 руб)	
Алёна Степанова	25 (12 796 000 руб)	2 (4 631 000 руб)	25 (12 796 000 руб)	
Иван Банов	7 (11 795 000 руб)	3 (3 025 000 руб)	7 (11 795 000 руб)	

Рисунок 14 – Аналитика AmoCRM

Из описания видно, что AmoCRM также является полнофункциональной системой и ничем не уступает Битрикс-24.

Из рассмотрения двух CRM-систем Битрикс-24 и AmoCRM можно сделать вывод о том, что наиболее востребованным является формат веб-приложений. Составим сравнительную таблицу для рассмотренных систем (таблица 1).

Таблица 1 - Сравнительный анализ CRM-систем

Критерии	Битрикс24	AmoCRM
Ведение этапов сделки	5	5
Телефония	5	5
Настройка под индивидуальные задачи	3	3
Аналитика и отчётность	5	5
Оплата по ежемесячной подписке	3	3
Регистрация клиента и управление данными	5	5
Регистрация менеджера и управление данными	5	5
Доступный интерфейс	3	3
Итого:	34	34

Из таблицы 1 можно увидеть, что рассмотренные CRM-системы по приведенным критериям оценки обладают огромным и, в случае применения их в отделе ДБО, избыточным функционалом, требуют детального изучения для начала работы, а значит, дополнительных ресурсов работников. Использование данных систем также является платным по ежемесячной, полугодовой или годовой подписке, что в перспективе приведёт к переплатам за пользование. Исходя из этого, целесообразным будет применить следующее решение – проектирование собственной CRM с учётом задач отдела ДБО с использованием веб-технологий.

Разрабатываемая система должна обеспечивать выполнение следующего функционала:

- учёт данных всех клиентов и их заявок, каналы продаж, соцсети,
- журнал действий, выполняемых специалистами ДБО,
- анализ деятельности и продуктивности специалистов,
- удалённая работа,
- авторизация и регистрация,
- аналитика финансовых показателей, начисление премий специалистам,
- учёт сделок.

Вывод по главе 1

В первой главе данной работы была рассмотрена организационная структура ПАО Банк «Александровский» и отдела ДБО, проанализированы бизнес-процессы отдела ДБО. В результате анализа были выявлены недостатки ведения бизнес-процессов, построены контекстные диаграммы IDEF0 «как есть» и «как должно быть». Исходя из определения потребностей в автоматизации бизнес-процессов отдела ДБО был проведён сравнительный анализ существующих решений на рынке CRM-систем для малого и среднего бизнеса в результате чего было принято решение о разработке CRM-системы под задачи отдела ДБО.

Глава 2 Проектная часть

2.1 Описание информационных потоков и событий системы

Разрабатываемая CRM будет представлять собой закрытую систему с разграничением прав доступа. Чтобы получить доступ к системе, необходимо пройти процесс авторизации с помощью ввода логина и пароля. При авторизации необходимо указать роль. Роли будут определены как «Администратор» и «Сотрудник» (специалист).

Система предназначена прежде всего для автоматизации и документирования всех действий специалистов. Каждый из авторизованных специалистов в системе будет иметь личный кабинет и свою клиентскую базу. В личном кабинете специалист сможет регистрировать клиента банка и вести клиента последовательно от процесса регистрации до процесса подключения к интернет-банкингу. Данные клиента, зарегистрированного специалистом, могут быть изменены данным специалистом. Таким образом, специалисту в личном кабинете будет доступен список клиентов, с которыми он работает. Также в личном кабинете специалиста будет доступна панель управления сделками. В зависимости от этапа прохождения сделки специалисту будут доступны те или иные действия. Сделка будет считаться завершённой, когда клиент получит статус «подключён». Данная сделка будет занесена в список успешных и отразится при расчёте аналитических показателей.

Если по каким-то причинам клиент откажется от услуг банка, то специалист сможет деактивировать его в системе.

Администратору системы будут доступны возможности добавления нового сотрудника с указанием всех его учётных данных, включая логин и пароль к системе. Данные логин и пароль администратор должен выдать для работы вновь зарегистрированному специалисту. Администратору доступны все данные о действиях сотрудников и их сделках, клиентская база и аналитическая панель расчёта бизнес-показателей.

Исходя из поставленных задач, на разработку CRM необходимо выделить рабочие процессы и информационные потоки (таблица 2).

Таблица 2 - События и потоки проектируемой CRM

Действующее лицо	Событие	Результат
Администратор	Авторизация	Вход в систему с правами администратора
	Регистрация нового сотрудника	В систему добавлен новый сотрудник
	Корректировка данных сотрудника	Данные по сотруднику скорректированы
	Удаление сотрудника из системы	Сотруднику закрыт доступ в систему
	Просмотр журнала событий	Просмотр всех доступных событий в системе
	Фильтрация журнала событий	Просмотр отфильтрованных по различным критериям событий
	Просмотр аналитики	Просмотр рассчитанных аналитических данных по сделкам специалистов и прочим метрикам
Специалист	Регистрация нового клиента	В систему добавлен новый клиент
	Корректировка данных клиента	Данные по клиенту скорректированы
	Удаление клиента из системы	Данные по клиенту перенесены в архив
	Ведение сделок	Продвижение по этапам сделки с помощью специальной панели
	Сортировка сделок	Упорядочивание сделок в личном кабинете с помощью сортировки и фильтрации.

Для отображения описанного выше функционала разрабатываемой системы необходимо построить диаграмму вариантов использования (прецедентов). Данная диаграмма демонстрирует основные варианты использования системы, а также возможные действия пользователя в той или иной роли (рисунок 15).

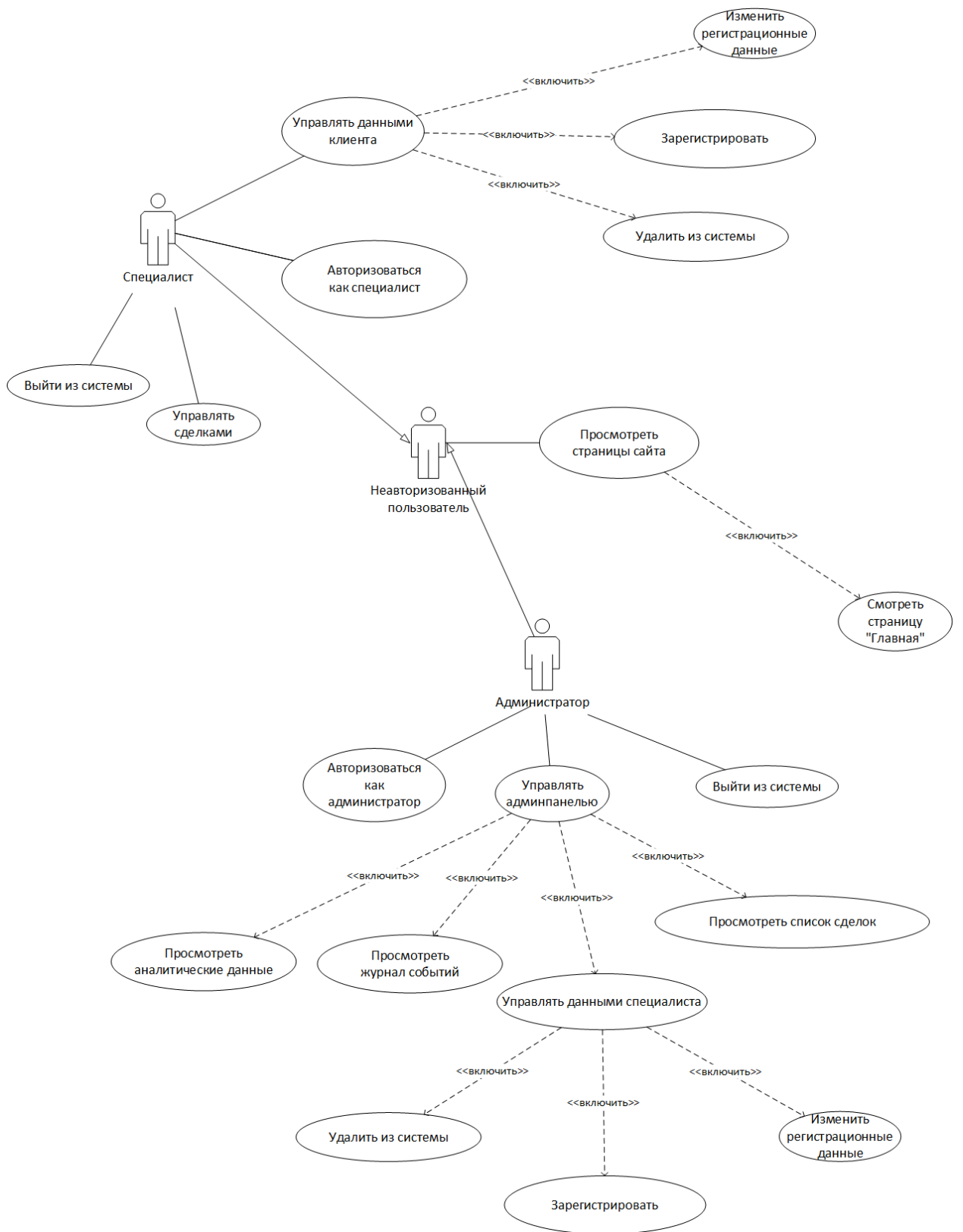


Рисунок 15 – Диаграмма вариантов использования системы

Диаграмма на рисунке 15 наглядно демонстрирует всех акторов системы и варианты действий, доступные им, что позволяет приступить к

непосредственному проектированию программной части разрабатываемой системы.

2.2 Выбор технологий и инструментов реализации

Исходя из постановки задачи, проектируемая CRM должна быть реализована с применением веб-технологий и возможностью выгрузки на удалённый хостинг. Проектируемую CRM-систему в дальнейшем по тексту можно считать веб-сайтом.

Как и большинство веб-приложений и сайтов, сайт CRM-системы имеет клиент-серверную архитектуру. Это означает, что необходимо обосновать выбор технологий, на базе которых проектируется клиентская и серверная часть. Существует ряд технологий, используемых при создании приложений (вебсайтов) по умолчанию, то есть в каждом случае, а именно – HTML, CSS и JavaScript [3]. Без данных языков разработка CRM-системы не представляется возможной. Минимально необходимым со стороны фронтенд-части является HTML как язык разметки, однако с помощью HTML возможно создать лишь контентную часть без оформления и без необходимых эффектов.

Стоит особо отметить, что рассматриваемые далее технологии и языки для разработки описаны с точки зрения их применимости в данном проекте, а не в отрыве от него. То есть всё описанное ниже: теги, атрибуты, классы, методы и так далее по большей части использованы в данной работе.

Для оформления вёрстки и улучшения внешнего вида CRM-системы в данной работе будет применён CSS-фреймворк «Bulma» и иконочный шрифт «Font Awesome». Рассмотрим их подробнее.

Преимущества Bulma:

- обеспечение адаптивного дизайна для настольных компьютеров, планшетов и мобильных телефонов,
- использование минимального HTML-кода, что позволяет легко читать и писать код.

Фреймворк Vulma предоставляет собой адаптивную сеточную систему (создающую макет страницы через ряд строк и столбцов), который масштабируется до двенадцати столбцов в контейнере.

Кроме этого, данный фреймворк поставляется с предварительно стилизованными компонентами, которые предоставляют выпадающие меню, модальные окна, нумерацию страниц, панель навигации, вкладки с целью облегчения работы пользователя на странице. Таким образом, управлять внешним видом CRM-системы при использовании Vulma станет гораздо легче, для этого необходимо присваивать элементам страницы соответствующие классы или использовать готовые компоненты.

В отличие от клиентской части, где реализация CRM-системы использует HTML и JavaScript, а различия могут быть лишь в применении различных библиотек, бэкенд часть может быть реализована с использованием различных языков программирования и систем управления базами данных. Выбор в данном сегменте довольно многообразен – C#, Python или PHP для программирования и MySQL, MicrosoftSqlServer, PostgreSQL для СУБД. Необходимо выбрать наиболее популярный стек разработки с наименьшим порогом входа.

При выборе языка программирования необходимо ориентироваться прежде всего на возможность полноценной разработки качественного сервиса. Если рассматривать C#, то данный язык разрабатывается и поддерживается компанией Microsoft, что несомненно может считаться стабильным решением. На C#, как правило, разрабатываются большие сложные приложения в формате «desktop» и в меньшей степени – веб-приложения, разработка на C# считается более дорогостоящей, нежели на языках программирования, предназначенных для веб-разработки. На языке Python также возможна полноценная веб-разработка, например, фреймворк Django позволяет создавать веб-сайты различной сложности, однако настройка Django требует более высокой квалификации программиста.

Скорость работы веб приложений на Python пока что не позволяет конкурировать по скорости выполнения с приложениями на PHP, который был изначально создан для проектирования веб-сайтов.

На PHP написано большинство сайтов в современном интернете [12], это означает огромную поддержку разработчиками и хорошую документацию всех функций.

Если делать выбор между базами данных, то необходимо рассмотреть наиболее популярные в сегменте веб-разработки – MySQL и PostgreSQL. СУБД во многом схожи по синтаксису и функционалу, однако PostgreSQL при этом имеет более продвинутый интерфейс и набор функций. Однако ключевую роль при выборе играет возможность настройки на сервере без дополнительных усилий при сохранении всей функциональности диалекта SQL.

Таким выбором безусловно станет PHP и MySQL как наиболее зарекомендовавшие себя в веб разработке. Данный стек имеет нативную поддержку у всех хостинг-провайдеров, что позволит легче интегрировать сервис в сети интернет в случае необходимости. К тому же с использованием данных технологий уже созданы многие системы корпоративного уровня, например, «ВКонтакте» или «Фейсбук».

MySQL представляет собой свободную реляционную систему управления базами данных. PHP очень часто используется в связке именно с MySQL, что позволяет воспользоваться преимуществами готовых решений этой связки, а также отработанными надёжными связями.

При разработке CRM-системы также необходимо использовать локальный сервер (XAMPP, MAMP, Open Server). Принципиального различия между ними нет: каждый из них обеспечивает взаимодействие PHP и MySQL. Выберем для целей разработки Open Server (рисунок 16) как наиболее популярный, а также предоставляющий по умолчанию приложение phpMyAdmin, умение работать с которым поможет легко интегрировать сайт на реальный хостинг. Использование Open Server позволяет провести весь

цикл разработки локально на компьютере разработчика, а затем выложить готовый исходный код на удалённый хостинг с минимальными изменениями.

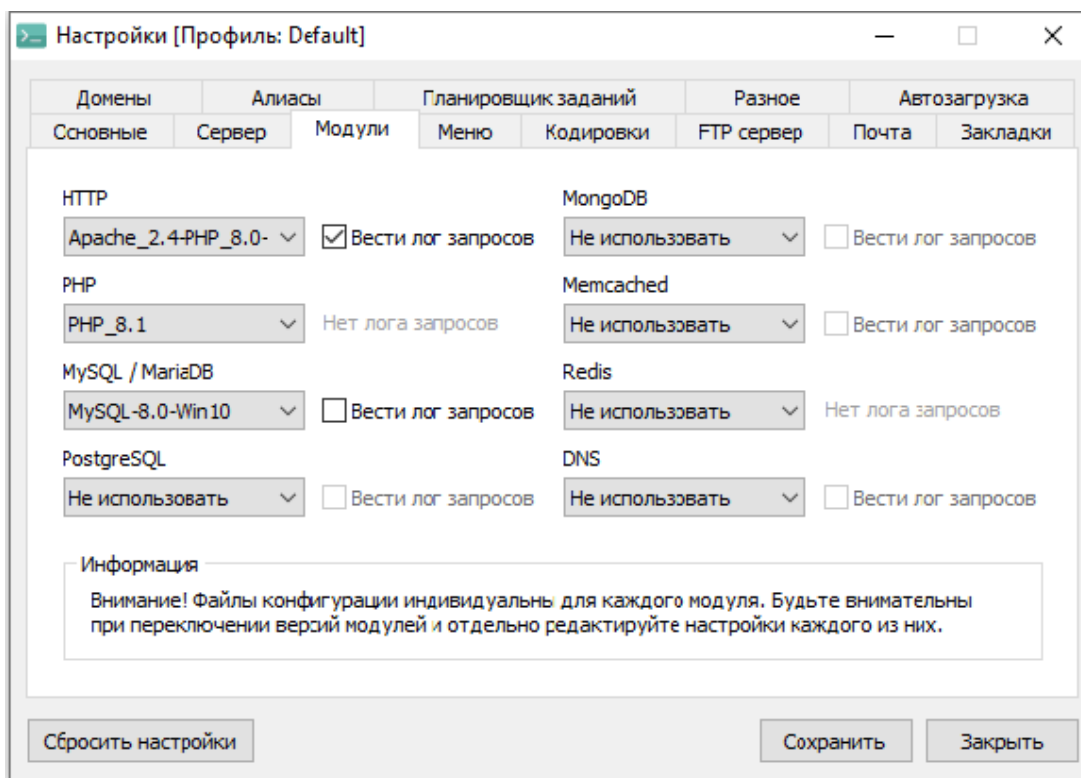


Рисунок 16 – Конфигурация Open Server для разработки

В качестве редактора для написания кода выберем Visual Studio Code. Данный редактор является бесплатным продуктом от компании Microsoft и пользуется большой популярностью среди разработчиков благодаря подключаемым плагинам, большинство из которых также бесплатны.

Выбранные технологии, языки программирования и инструменты разработки поместим в сводную таблицу. (таблица 3)

Таблица 3 - Набор выбранных инструментов и технологий для проектирования

Функция	Название	Описание
Система управления базами данных	MySQL	Хранение, обработка и целостность данных.

Продолжение таблицы 3

Функция	Название	Описание
Отрисовка сайта системы	HTML + CSS	Внешний вид сайта в браузере
Взаимодействие с элементами веб - страниц	JavaScript	Интерактивное взаимодействие
Логика серверной части и взаимодействие с базой данных	PHP	Ядро программной части, расположенное на веб-сервере
Веб-сервер для локальной разработки	OpenServer	Программная оболочка серверной части сайта
Среда разработки	Visual Studio Code	Редактор кода

Предложенный в таблице 3 набор технологий и средств проектирования позволяет создать полноценное многофункциональное веб-приложение, способное решать задачи различного уровня сложности.

2.3 Разработка базы данных

Физические сущности, задействованные в системе CRM, в последствии необходимо преобразовать в таблицы базы данных. Сущности представляют собой определённые объекты, действия, процессы или абстракции, которые участвуют в технологическом процессе работы CRM. У каждой сущности должны быть атрибуты – физические элементы, характеризующие каждую сущность. Необходимо выделить сущности и атрибуты сущностей, которые должны быть преобразованы в таблицы. Исходя из требуемого функционала и методов его реализации, можно выделить сущности. (таблица 4)

Таблица 4 - Сущности базы данных

Сущности	Назначение
Журнал	Данные о действиях специалистов в системе
Администраторы	Данные об администраторах системы
Сотрудники	Данные о сотрудниках
Клиенты	Данные о клиентах
Сделки	Данные о сделках
Каналы	Данные о каналах привлечения клиентов

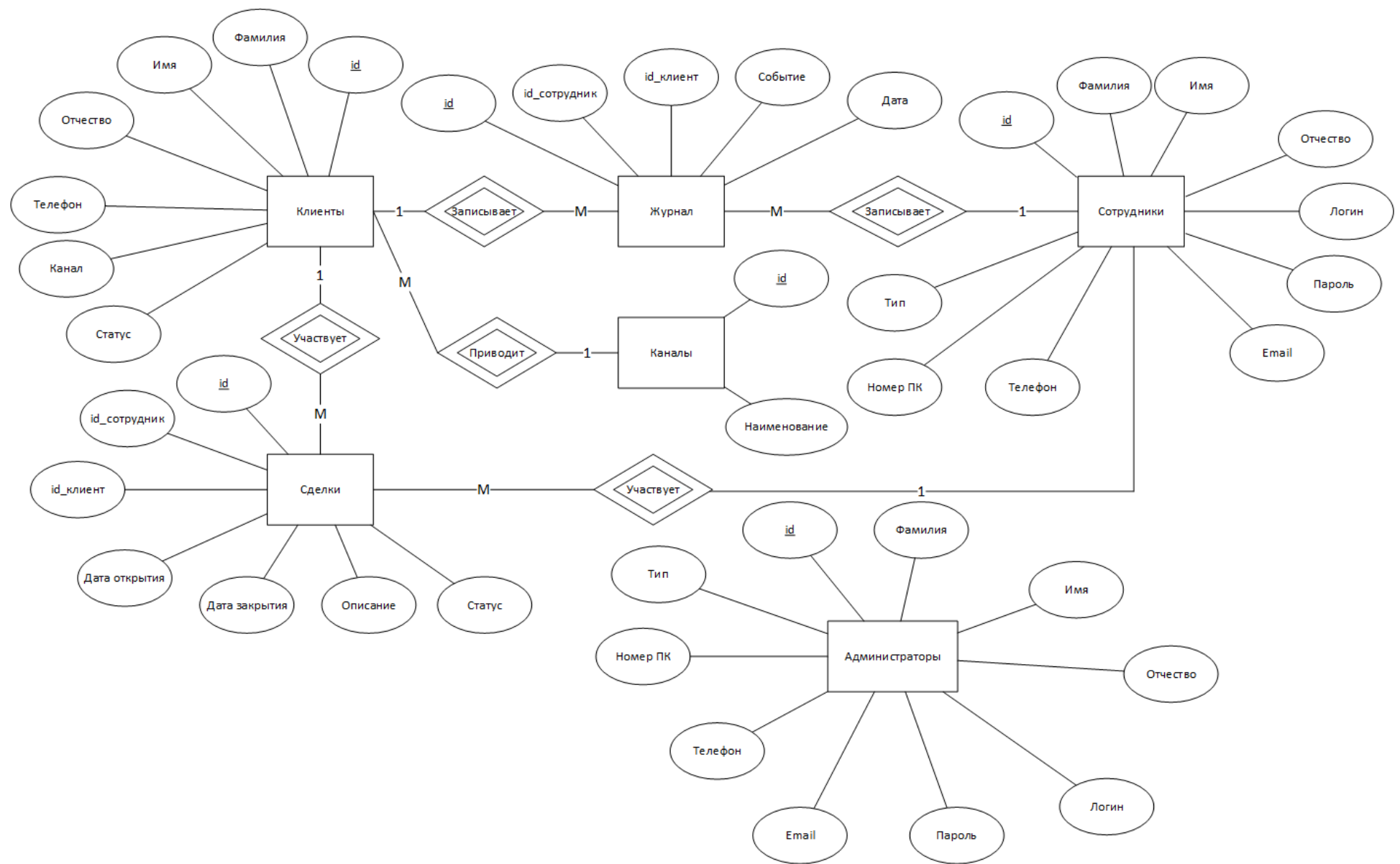


Рисунок 17 – Инфологическая схема базы данных

У каждой сущности должны быть атрибуты – физические элементы, характеризующие каждую сущность. Сущности, их атрибуты и связи между ними отобразим в виде инфологической модели базы данных (рисунок 16).

В качестве ограничений целостности базы данных будут использованы следующие типы ключей – первичный, внешний. Также каждый из атрибутов будет иметь характерный тип данных, ограниченный по длине.

При дальнейшем проектировании базы данных необходимо построить ER-диаграмму (рисунок 18), отображающую таблицы сущностей с атрибутами, связи между ними, а также внешние и первичные ключи.

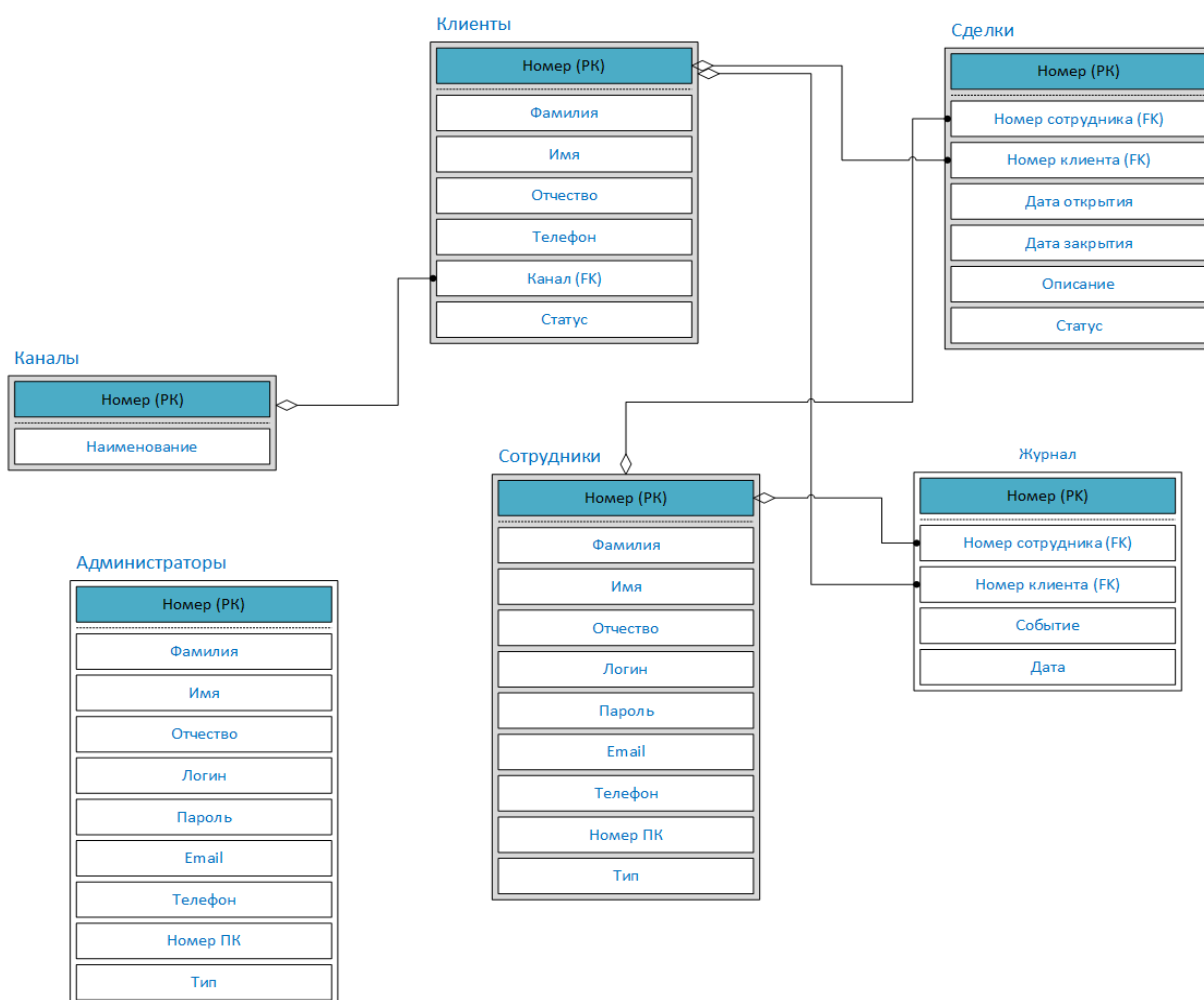


Рисунок 18 – ER диаграмма базы данных

Данные таблицы соответствуют требованиям нормализации реляционных БД. Таблицы находятся в 1НФ (нормальная форма), так как все их поля являются простыми (неделимыми), нет повторяющихся полей и каждый столбец таблиц хранит одно-единственное значение и не является ни списком, ни множеством значений. Таблицы находятся во 2НФ, так как они находятся в 1НФ, имеют простые первичные ключи и каждое не ключевое поле функционально зависит от первичного ключа (нет составных ключей). Таблицы находятся в 3НФ, так как они находятся во 2НФ и все не ключевые поля являются взаимно-независимыми

Для данных, связанных со временем, используем тип «Datetime», для числовых данных – тип «INT» и «TINYINT», для вариантов выбора – «ENUM», для текстовых данных – «VARCHAR». Исходя из этого необходимо составить набор отдельных сущностей базы данных в виде таблиц.

В результате выполнения SQL – кода (Приложение А) была получена база данных, схема которой показана на рисунке 19.

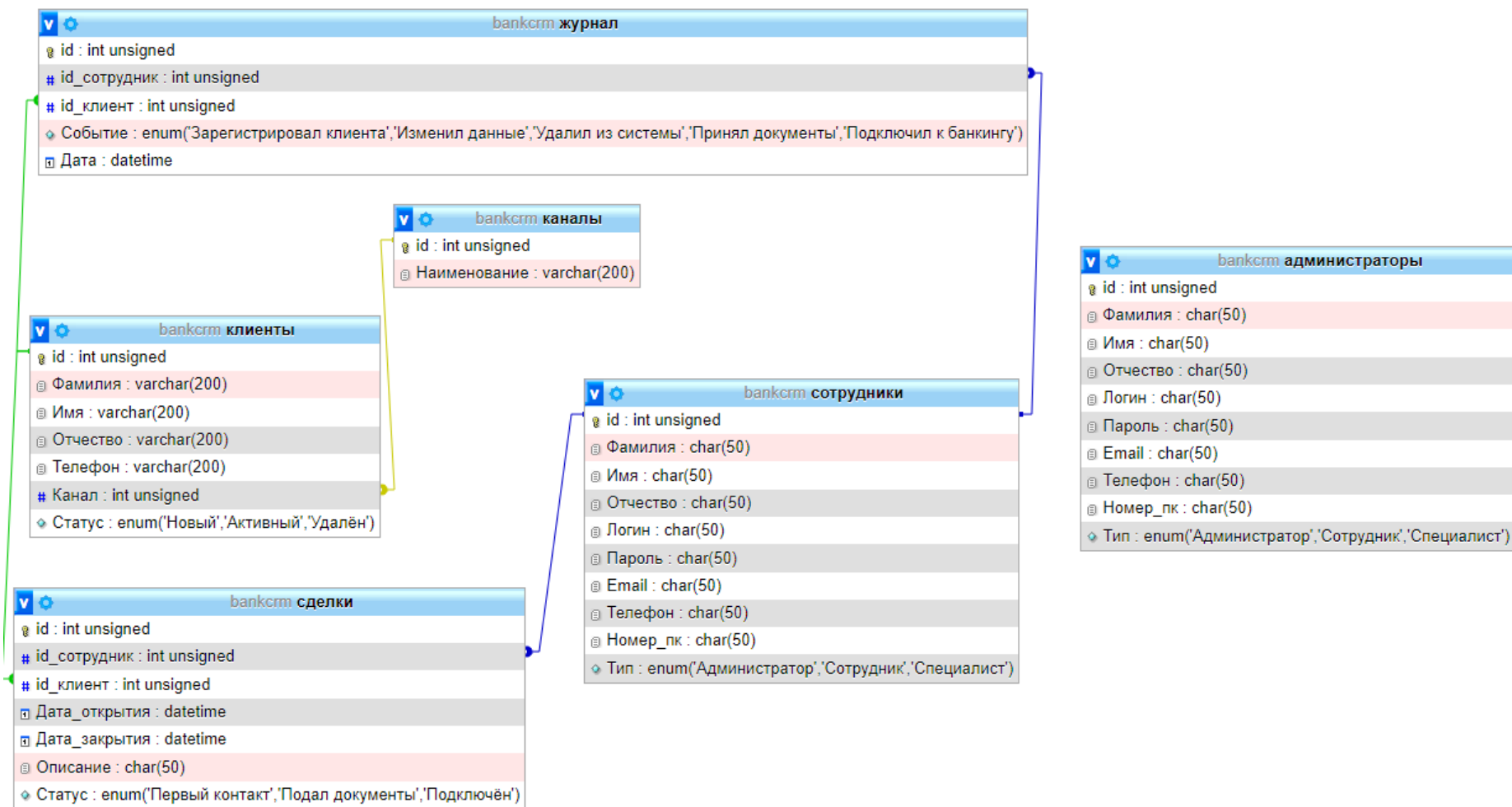


Рисунок 19 – Физическая схема базы данных

2.4 Разработка интерфейса и фронтенд-части

При дальнейшем проектировании CRM-системы нужно создать описание пользовательского интерфейса, определить все необходимые для реализации функционала элементы.

Основным критерием внешнего вида будет корректное отображение страниц в современных браузерах.

Код представлений (видов) страниц содержится в папке «templates» проекта. Каждый файл представлений страниц подключается соответствующим, как правило, одноимённым, контроллером.

В общем виде, контроллер страницы собирает данные посредством модели, затем формирует отдельные фрагменты кода HTML из шаблонов «header.php», «main.php», «footer.php» и вставляет их в шаблон «layout.php».

На рисунке 20 расположено изображение макета страниц системы, каждый элемент макета соответствует файлу php: шаблон (layout.php), верхняя часть (header.php), нижняя часть (footer.php), контент (main.php).



Рисунок 20 – Макет страниц сайта

Такой подход имеет ряд преимуществ и позволит использовать готовые компоненты кода в приложении. Достаточно прописать статическую часть кода в данных компонентах и подключать их на различных страницах. Это позволит избежать дублирование кода в разных местах.

В работе выбран компонентный подход к формированию веб-интерфейса, в результате чего был разработан макет страниц сайта, что позволит разделить код на фрагменты и подключать их отдельные части в необходимых местах.

2.5 Разработка серверной части

При проектировании сайта системы необходимо учесть возможность его дальнейшего развития [11]. Чтобы учесть возможные будущие требования к развитию системы, необходимо использовать модульный подход к разработке и паттерны проектирования, в частности при разработке веб-интерфейса необходимо использовать заранее определённые наборы классов для оформления внешнего вида страниц, что позволит в дальнейшем добавлять элементы интерфейса и стилизовать их заранее подготовленными классами. Со стороны бизнес-логики также необходимо организовать архитектуру приложения со стороны сервера, используя для этого паттерн MVC (model-view-controller), который будет описан далее, и который позволит разделить логику и внешний вид, а также разбить весь код на модули. При появлении нового функционала и новых страниц можно будет добавлять новые модули, следуя правилам MVC, что позволит упростить внедрение нового кода, а также существенно упростит поддержку существующего [17].

Все виды (views) страниц сайта будут размещены в отдельной папке «templates», контроллеры будут иметь одноимённые названия со страницами сайта и выполнять все действия бизнес-логики, моделью станет файл «functions.php». Набор функций модели будет содержать в себе функции

обращения к базе данных, работы с шаблонами, работы с файлами cookies и прочие вспомогательные функции.

Таким образом, контроллер отдельной страницы будет подключать модель, затем формировать переменные и предавать их в вид, формировать этот вид и выводить полученный результат пользователю.

Схема взаимодействия модулей представлена на рисунке 21

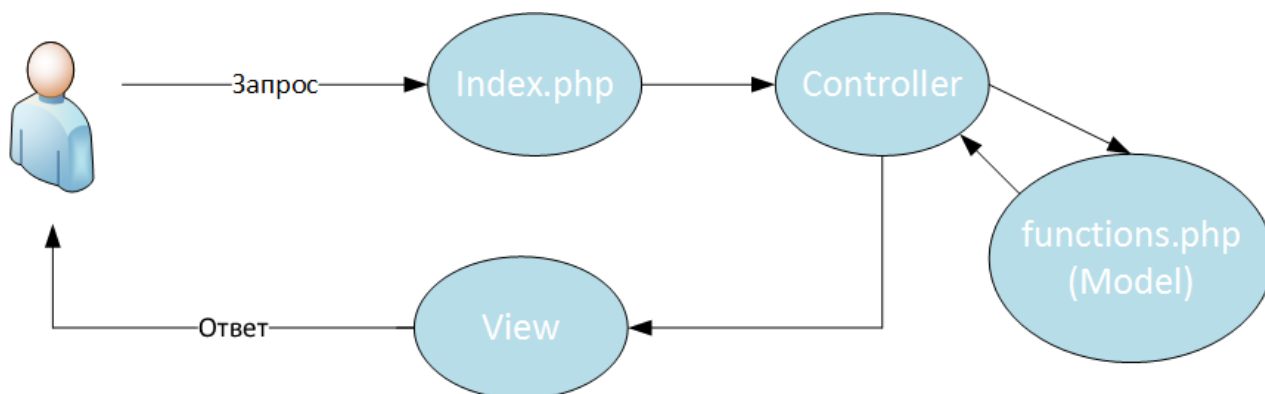


Рисунок 21 – Применение паттерна MVC в проекте

Рассмотрим основные функции, которые будут использоваться в данном проекте. Функции расположены в файле functions.php. Данные функции написаны самостоятельно, исходя из логики работы проектируемого сайта системы CRM.

Работа с базой данных основывается на встроенном в PHP классе PDO, который предоставляет все необходимые для этого методы и свойства. Подключение к базе данных инициализируется созданием экземпляра класса PDO с передачей в конструктор объекта параметров подключения, таких как наименование драйвера базы данных, URL адрес хоста подключения, имя базы данных, логин и пароль. На основе полученного экземпляра при выполнении различных методов класса PDO происходит работа с базой данных. Одним из основных методов класса PDO является метод query, аргументом которого может быть текст в виде любого SQL-запроса к базе данных, результатом исполнения данного метода является экземпляр класса PDOStatement, методы

которого позволяют непосредственно извлекать полученные данные из базы. Примерами методов PDOStatement являются fetch() и fetchAll(), которые извлекают очередную строку из результирующего набора, либо все строки сразу. Строки могут быть размещены в виде объекта или в виде массива, в зависимости от настроек PDO. Реализованные в данной работе функции dbGo(), sql(), sqlInsert() представляют собой обёртки, реализующие поведение вышеописанных классов и методов.

dbGo() – иницирует подключение к базе данных и возвращает экземпляр класса PDO в случае успешного подключения, а в случае сбоя при подключении в поток вывода выдаётся информация об ошибке и скрипт завершается. Листинг функции dbGo() представлен в приложении Б.

sql(\$q) – запускает внутри себя dbGo(), а затем исполняет переданный в неё в качестве аргумента SQL запрос (\$q), возвращая в виде результата экземпляр PDOStatement, к которому в зависимости от задачи можно применить методы fetch() или fetchAll(). Листинг функции sql() представлен в приложении В.

sqlInsert() позволяет вставить данные в таблицу, использует в качестве аргументов два параметра: наименование таблицы и массив данных. Внутри данной функции происходит обработка переданных данных для последующего формирования SQL запроса, затем происходит вызов dbGo(), к результату вызова которой применяется метод exec(), в который передаётся сформированный ранее SQL-запрос. Результат исполнения этого метода возвращает сама функция sqlInsert(), это может быть 1 или 0, сигнализирующие о том, успешно или нет прошла вставка данных в таблицу. Листинг функции sqlInset() представлен в приложении Г.

Так как приложения на PHP после выполнения различных действий с данными пользователя, базы данных или внешних сервисов в конечном счёте возвращают пользователю HTML-страницу, то встаёт вопрос о компоновке HTML-кода и работе с этим кодом самого PHP. PHP необходимо сгенерировать разметку, предварительно передав туда какие-либо параметры,

полученные в результате работы скрипта. Причём сами параметры могут представлять собой уже ранее сгенерированные фрагменты кода, что даёт очень большую гибкость в управлении внешним видом получаемой страницы. Именно на этом принципе строится шаблонизация: PHP-скрипт собирает мелкие части разметки и вставляет их в более крупные, затем эти более крупные вставляет в страницу и формирует окончательный вид, отправляя её в браузер. В данной работе также используются принципы шаблонизации, идеи которых реализует написанная функция `template($view, $arr, $path)`. Листинг функции `template()` представлен в приложении Д

`template($view, $arr, $path)` принимает на вход три параметра: `$view` – название подключаемого вида, `$arr` – массив с данными, `$path` – путь к файлу директории. При применении встроенной в PHP функции `extract()` происходит распаковка массива `$arr`, в результате чего образуются переменные, далее включается буферизация вывода с помощью встроенной в PHP функции `ob_start()`, на основе параметров `$view` и `$path` формируется путь к подключаемому шаблону, в результате чего шаблон подключается с помощью конструкции `include()`, PHP-интерпретатор считывает данный шаблон, накапливая в буфере текст и подставляя в него переменные. По окончании считывания шаблона происходит возврат из функции `template()`, результатом работы которой является сброс в переменную накопленного и обработанного HTML-текста.

Вывод по главе 2

Во второй главе данной работы было произведено проектирование системы CRM. В результате анализа информационных потоков и построения диаграммы прецедентов были сформированы сущности для проектирования базы данных, на основе которых построена физическая схема базы данных. Спроектирован интерфейс внешнего вида системы, а также разработаны скрипты серверной части и скрипты фронтенд-части. Таким образом, проектирование системы полностью завершено.

Глава 3 Анализ работы CRM

3.1 Контрольный пример использования

В системе предусмотрено две роли – «администратор» и «специалист», каждый из них имеет доступ к личному кабинету. В личном кабинете администратора доступен следующий функционал: добавление, изменение и удаление данных о специалистах, просмотр журнала событий, просмотр аналитических данных результатов работы. В личном кабинете специалиста доступен следующий функционал: добавление, изменение и удаление данных о клиентах, добавление и ведение сделок.

Старт программы начинается с окна авторизации (рисунок 22). Авторизоваться можно в роли «администратор» или «специалист». Произведём авторизацию в роли «администратор» с помощью логина и пароля.

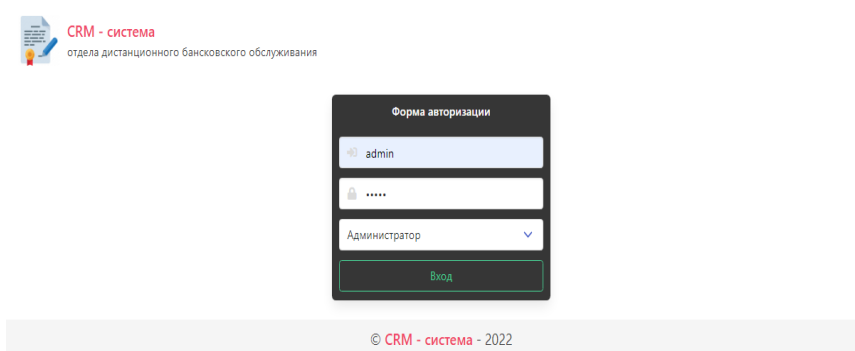


Рисунок 22 – Страница запуска программы

После авторизации в роли «администратор» будет показана страница навигации по личному кабинету с общей информацией о возможностях роли «администратор» и ссылками на страницы с соответствующим функционалом (рисунок 23).

Администратору доступен расширенный функционал системы.
Администратор может управлять списком сотрудников - добавлять, удалять и редактировать информацию о них.
Администратору доступен список сделок с клиентами ДБО и расширенная аналитика по ним.

Рисунок 23 – Личный кабинет администратора

В разделе «персонал» располагается список сотрудников (рисунок 24) и форма для добавления в систему нового сотрудника (рисунок 25). Список сотрудников оформлен в виде таблицы с изменяемыми полями. В каждое поле могут быть внесены изменения, а затем сохранены в базу данных. Также существует возможность удалить учётную запись сотрудника из системы.

СПИСОК СОТРУДНИКОВ

Фамилия	Имя	Отчество	Email	Телефон	Номер ПК	Действие	
Иванов	Олег	Николаевич				Изменить	Удалить

Рисунок 24 – Управление данными сотрудников

Добавить сотрудника

Фамилия

Имя

Отчество

Тип

Логин

Пароль

Добавить

Рисунок 25 – Форма добавления нового сотрудника

После заполнения формы и отправки данных на сервер в списке сотрудников будет добавлена вновь созданная учётная запись (рисунок 26). При добавлении новой учётной записи сотрудника автоматически генерируются логин и пароль, которые должны быть переданы администратором системы новому сотруднику.

СПИСОК СОТРУДНИКОВ

Фамилия	Имя	Отчество	Email	Телефон	Номер ПК	Действие	
Максимов	Олег	Сергеевич	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Изменить"/>	<input type="button" value="Удалить"/>
Иванов	Александр	Сергеевич	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Изменить"/>	<input type="button" value="Удалить"/>

Рисунок 26 – Список сотрудников

Сотрудник, имеющий в распоряжении данные своей учётной записи, имеет доступ к личному кабинету и может управлять данными клиентов

(регистрация, изменение, архивирование в системе), а также вести сделки по подключению клиентов к интернет-банкингу.

Каждый сотрудник имеет собственную клиентскую базу, закреплённую за ним (рисунок 27).

CRM - система
отдела дистанционного банковского обслуживания

Клиенты Сделки Выход

Авторизован: Иванов О.
Роль: Специалист

КЛИЕНТСКАЯ БАЗА

Номер	Описание	Дата создания	Статус	Документ	Действие
5	Иванов	Александр	Игоревич	+79877896555	Изменить Удалить
6	Борисова	Надежда	Сергеевна	+79877896551	Изменить Удалить
4	Максимов	Андрей	Иванович	+7987789655	Клиент был удален из системы

Рисунок 27 – Личный кабинет специалиста

После изменений данных клиента сотрудник получит уведомление о совершённом действии.

Это сделано при помощи флеш-сообщений, хранящихся в сессии PHP (рисунок 28).

Данные были изменены

КЛИЕНТСКАЯ БАЗА

Номер	Описание	Дата создания	Статус	Документ	Действие
5	Иванов	Александр	Игоревич	+79877896555	Изменить Удалить
6	Борисова	Надежда	Сергеевна	+79877896551	Изменить Удалить
4	Максимов	Андрей	Иванович	+7987789655	Клиент был удален из системы

Рисунок 28 – Изменение информации о клиентах

В разделе «сделки» представлена панель управления процессом подключения клиента к интернет-банкингу. Также список сделок может быть отсортирован по критериям «статус», «дата», «тип» (рисунок 29).

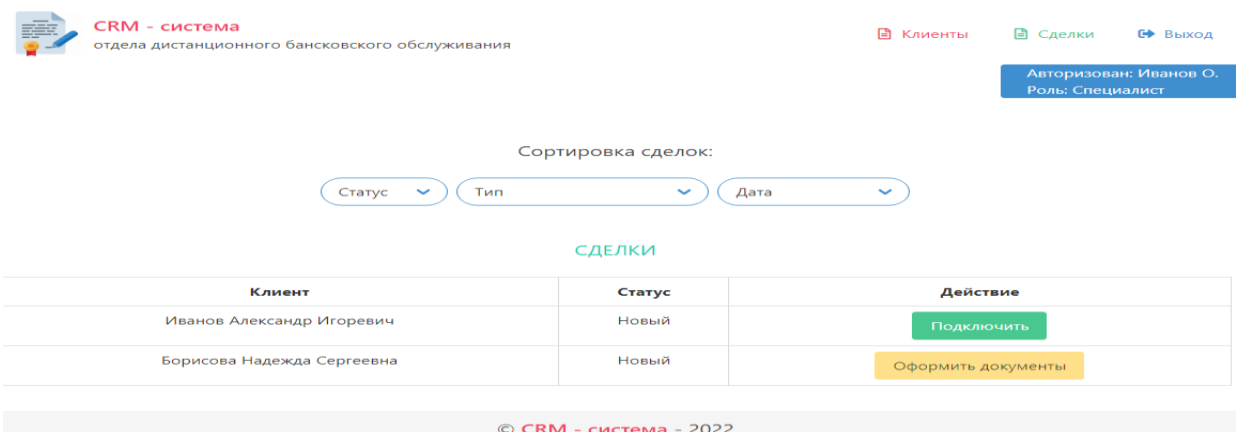


Рисунок 29 – Панель управления сделками

Чтобы добавить нового клиента в систему, сотруднику необходимо заполнить форму и отправить данные на сервер (рисунок 30), после чего запись клиента появится в списке доступных (рисунок 31).

Новый клиент

Фамилия
Александрова

Имя
Марина

Отчество
Николаевна

Телефон
+79877896552

Канал
Telegram

Добавить

Рисунок 30 – Форма добавления нового клиента

Клиент добавлен в систему

КЛИЕНТСКАЯ БАЗА

Номер	Описание	Дата создания	Статус	Документ	Действие
12	Александрова	Марина	Николаевна	+71877896552	Изменить Удалить

Рисунок 31 – Информация о вновь добавленном клиенте

Действия, произведённые каждым из сотрудников (специалистов) в своём личном кабинете, записываются в журнал событий (рисунок 32), доступный администратору системы. К таким действиям относятся события по этапам сделки и события, связанные с управлением данными клиентов.

CRM - система
отдела дистанционного банковского обслуживания

Журнал Сделки Персонал Аналитика Выход

Авторизован: Макаров А.
Роль: Администратор

ЖУРНАЛ СОБЫТИЙ

Клиент	Статус сделки	Дата
Максимов Олег Сергеевич	Зарегистрировал клиента	2022-10-09 15:47:47
Максимов Олег Сергеевич	Принял документы	2022-10-09 15:58:05
Максимов Олег Сергеевич	Подключил к банку	2022-10-09 16:00:44
Максимов Олег Сергеевич	Зарегистрировал клиента	2022-10-09 17:07:12
Максимов Олег Сергеевич	Принял документы	2022-10-09 17:07:17
Максимов Олег Сергеевич	Подключил к банку	2022-10-09 17:07:17
Максимов Олег Сергеевич	Зарегистрировал клиента	2022-10-09 20:04:42
Максимов Олег Сергеевич	Принял документы	2022-10-09 20:04:45
Максимов Олег Сергеевич	Подключил к банку	2022-10-09 20:04:46
Белик Ирина Викторовна	Изменил данные	2022-10-09 16:57:30

Рисунок 32 – Журнал событий

В разделе «аналитика» личного кабинета роли «администратор» (рисунок 33) доступны вычисляемые данные о завершённых сделках, то есть сделках, имеющих статус «подключен». Данные формируются динамически в зависимости от выбранного месяца и представлены в виде гистограммы. Гистограмма отображает количество завершённых сделок тем или иным сотрудником (специалистом). Данный рисунок формируется при каждом новом запросе с помощью PHP и встроенной графической библиотеки GD2.

Выберите период:

Октябрь 2022 

Номер	Дата открытия	Статус
49	2022-10-09 15:47:47	Подключён
50	2022-10-09 17:01:40	Подключён
51	2022-10-09 17:07:12	Подключён
52	2022-10-09 20:02:57	Подключён
53	2022-10-09 20:04:42	Подключён



Рисунок 33 – Страница с аналитическими данными

В контрольном примере был рассмотрен вариант работы спроектированной CRM-системы. В результате выполнения различных действий как в роли «администратор», так и в роли «специалист» ошибок выявлено не было. Система работает корректно.

3.2 Расчёт экономической эффективности

В данном разделе необходимо произвести расчёт затрат на разработку CRM, и впоследствии рассчитать экономический эффект от применения разработки, тем самым обосновав экономическую целесообразность.

Составим смету на расходы, пункты в которой будут следующие:

- стоимость оборудования и программного обеспечения;
- стоимость расходных материалов;
- основная и дополнительная заработная плата;
- обязательные платежи во внебюджетные фонды от основной и дополнительной заработной платы;
- накладные расходы;
- прочие расходы.

Произведем расчет по каждому пункту в отдельности.

Таблица 5 - Перечень оборудования и программного обеспечения

Наименование	Количество, ед.	Стоимость единицы, руб.	Балансовая стоимость, руб.
Windows 10	1	6400	6400
Ноутбук	1	30000	30000
Итого			36400

Таблица 6 - Данные по расходам на материалы

Статьи затрат	Количество, ед.	Коэффициент использования, %	Стоимость, руб.	Сумма, руб.
Флэш-диск	1	90	380,00	480,00
Бумага офисная	2	80	210,00	420,00
Набор канцелярии	1	80	35,00	35,00
Мышка	1	90	400	
Итого				1335,00

В проектировании и разработке CRM принимали участие 2 человека.

Определим продолжительность выполнения работ (трудоемкость) проектирования и разработки CRM. Результаты представим в таблице 7.

Таблица 7 - Список работ по проектированию и разработке CRM

Наименование работ	Исполнитель	Трудоемкость, час
Изучение задания	Студент дипломник	5
	Научный руководитель	1
Сбор информации	Студент дипломник	13
	Научный руководитель	2
Анализ учебной и научной литературы	Студент дипломник	14
	Научный руководитель	2
Проектирование и разработка CRM	Студент дипломник	24
	Научный руководитель	12
Выводы по проделанной работе	Студент дипломник	7
	Научный руководитель	3
Итого	Студент дипломник	63
	Научный руководитель	20

Выполнение заняло 63 часа (1 месяц).

Работы, направленные на разработку информационной системы, выполняются техником-программистом.

Расчет основной заработной платы включает в себя:

- расчет оплаты выполненных работ по тарифу.
- расчет премиальных доплат.
- расчет прочих доплат.
- расчет доплат поясного коэффициента.

Расчет оплаты труда студенту.

Заработная плата по тарифу рассчитывается по формуле (1):

$$Зтар = Тр \cdot Тст, (\text{руб.}) \quad (1)$$

где Тр – трудоемкость всех видов работ по созданию CRM, час.

Тст – часовая тарифная ставка, руб.

$$Зтар = 63 \cdot 5,3 = 334 (\text{руб.})$$

Премиальные доплаты рассчитывают по формуле (2):

$$Дпрем = 0,5 \cdot Зтар, (\text{руб}) \quad (2)$$

$$Дпрем = 0,5 \cdot 334 = 167 (\text{руб})$$

Доплаты поясного коэффициента включают в себя 0,20 – районный коэффициент; 0,30 – процентная надбавка за стаж работы в районах крайнего севера и рассчитываются по формуле (3):

$$Дпк = 0,5 \cdot (Зтар + Дпрем), (\text{руб.}) \quad (3)$$

$$Дпк = 0,5 \cdot (334 + 167) = 251 (\text{руб.})$$

Основная заработная плата рассчитывается по формуле (4):

$$Зосн = Зтар + Дпрем + Дпк (\text{руб.}) \quad (4)$$

$$Зосн = 334 + 167 + 251 = 752 \text{ (руб)}$$

Расчет оплаты труда руководителю.

Заработная плата по тарифу рассчитывается по формуле (5):

$$Зтар = Тр \cdot Тст, \text{ (руб.)} \quad (5)$$

где Тр – трудоемкость всех видов работ по созданию CRM, час.

Тст – часовая тарифная ставка, руб.

$$Зтар = 20 \cdot 82,7 = 1654 \text{ (руб.)}$$

Премииальные доплаты рассчитывают по формуле (6):

$$Дпрем = 0,5 \cdot Зтар, \text{ (руб)} \quad (6)$$

$$Дпрем = 0,5 \cdot 1654 = 827 \text{ (руб.)}$$

Прочие доплаты рассчитываются по формуле (7):

$$Дпр = 0,1 \cdot Зтар, \text{ (руб.)} \quad (7)$$

$$Дпр = 0,1 \cdot 1654 = 165 \text{ (руб.)}$$

Доплаты поясного коэффициента рассчитываются по формуле (8):

$$Дпк = 0,5 \cdot (Зтар + Дпрем + Дпр), \text{ (руб)} \quad (8)$$

$$Дпк = 0,5 \cdot (1654 + 827 + 165) = 1323 \text{ (руб.)}$$

Основная заработная плата рассчитывается по формуле (9):

$$Зосн = Зтар + Дпрем + Дпр + Дпк, \text{ (руб)} \quad (9)$$

$$Зосн = 1654 + 827 + 165 + 1323 = 3969 \text{ (руб.)}$$

Заработная плата складывается из заработной платы студенту (стипендии) и оплаты работы руководителя дипломной работы.

Часовая тарифная ставка студенту дипломнику рассчитывается исходя из размера базовой стипендии. Прочие доплаты студенту дипломнику не предусмотрены.

Часовая тарифная ставка научного руководителя оценивается по отраслевой системе оплаты труда.

К дополнительной заработной плате относятся оплата отпусков, периодов нетрудоспособности работника и тому подобное. Дополнительная заработная плата укрупненно рассчитывается как 13,3 % от основной заработной платы (10):

$$З_{доп} = 0,133 \cdot З_{осн}, (\text{руб.}) \quad (10)$$

$$З_{доп} = 0,133 \cdot 3969 = 528 (\text{руб.})$$

Общая заработная плата студента и руководителя:

$$З_{общ} = 3969 + 752 = 4721 (\text{руб.})$$

Общая заработная плата студента и руководителя, а также дополнительная заработная плата:

$$З_{общ} = 4721 + 528 = 5249 (\text{руб.})$$

На основе Федерального закона РФ от 24.07.2009 №12 (в ред. от 02.04.2015) «О страховых взносах в Пенсионный фонд Российской Федерации, Фонд социального страхования Российской Федерации, Федеральный фонд обязательного медицинского страхования» отчисления составляют:

- в пенсионный фонд РФ – 22%.
- в фонд социального страхования РФ – 2,9%.
- в фонд обязательного медицинского страхования РФ – 5,1%.

На основе Федерального закона РФ от 21.12.2021 N 413-ФЗ «О страховых тарифах на обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний на 2022 год и на

плановый период 2023 и 2024 годов» отчисления составляют 0,2%. Стипендия студента-дипломника в фонде оплаты труда не учитывается (11):

$$\text{ОПВБФ} = 0,32 \cdot (\text{Зосн} + \text{Здоп}), (\text{руб.}) \quad (11)$$

$$\text{ОПВБФ} = 0,32 \cdot (5249) = 1680 (\text{руб.})$$

Накладные расходы включают в себя расходы на ремонт, амортизационные отчисления и электроэнергию, оплату консультаций, арендную плату и тому подобное.

Укрупненно накладные расходы принимаются в размере 100-120% от размера основной заработной платы.

$$\text{Рнакл} = (4721) \cdot 1,1 = 5193 (\text{руб.})$$

В группу затрат с прочими расходами включаются расходы на использование сети Интернет, телефонная связь, транспортные расходы и тому подобное.

Укрупненно прочие расходы составляют 5-10% от суммы всех затрат.

$$\text{Рпр} = (5249 + 1680 + 5193) \cdot 0,1 = 1212 (\text{руб.})$$

Рассчитанные затраты на проектирование и разработку CRM сводим в таблицу 8.

Таблица 8 - Рассчитанные затраты на проектирование и разработку сайта

Статья затрат	Сумма, руб.
Регистрация и право владения доменом	350
Хостинг	400
Оборудование и программное обеспечение	36400
Стоимость расходных материалов	1335
Основная и дополнительная заработная плата	5249
Обязательные платежи во внебюджетные фонды от основной и дополнительной заработной платы	1680
Накладные расходы	5193
Прочие расходы	1212
Итого:	51819

После внедрения CRM ожидается положительный экономический эффект, который связан с отсутствием различных утечек данных, оцениваемых в год порядка 15000 рублей. Также планируется увеличение продуктивности работы сотрудников, благодаря чему высвобождаются дополнительные часы рабочего времени, этот эффект можно оценить в 43000 рублей за один год. Таким образом, общий положительный эффект составит 58000 рублей за один год.

Срок окупаемости равен отношению затрат к положительному эффекту и составит $51819 / 58000 = 0.893$ лет, или примерно 10,7 месяцев. Исходя из этого, реализация CRM может считаться экономически оправданной и перспективной.

На основе данных из таблицы 8 можно построить круговую диаграмму, чтобы визуализировать результаты.

Затраты на проектирование и разработку

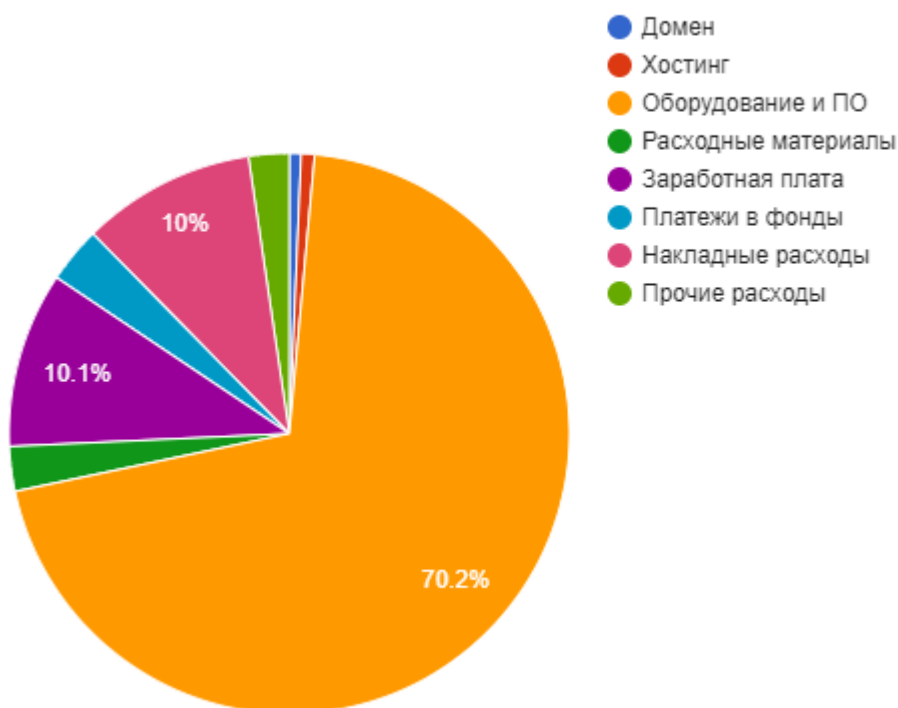


Рисунок 34 – Диаграмма затрат на проектирование и разработку CRM

Вывод по главе 3

В главе 3 представлен контрольный пример использования спроектированной CRM-системы. Личный кабинет CRM-системы представлен двумя ролями: «администратор» и «специалист». Последовательно выполнены шаги от авторизации в роли «администратор» до управления сделками в роли «специалист». «Администратор» создаёт учётную запись специалиста. «Специалист» вносит в CRM-систему сведения о клиентах. Полученные в результате этих действий данные продемонстрированы в «журнале событий» и разделе «аналитика». Журнал событий отображает информацию о том, какие действия в CRM-системе произвёл каждый сотрудник. В разделе «аналитика» приведена гистограмма, отображающая сравнительное количество завершённых сделок по всем сотрудникам. По результатам тестирования системы CRM ошибок обнаружено не было, что позволяет сделать выводы о том, что система работает корректно и в соответствии с заявленными требованиями. Также в главе 3 был произведён расчёт экономической эффективности, который показал, что разработка системы CRM была обоснованной и приведёт к высвобождению дополнительных финансовых средств после срока окупаемости равного 10,7 месяцам.

Заключение

В данной работе был произведён анализ деятельности отдела дистанционного банковского обслуживания ПАО Банк «Александровский» и разработана система CRM для нужд данного отдела.

В первой главе была рассмотрена организационная структура ПАО Банк «Александровский» и отдела ДБО, проанализированы бизнес-процессы отдела ДБО. В результате анализа были выявлены недостатки ведения бизнес-процессов, в первую очередь связанные с отсутствием рабочих инструментов по автоматизации бизнес-процессов. Исходя из определения потребностей в автоматизации бизнес-процессов отдела ДБО был проведён сравнительный анализ существующих решений на рынке CRM-систем для малого и среднего бизнеса. Анализ данных систем позволил учесть лучшие практики при построении подобных инструментов. В результате анализа с учётом всех факторов было принято решение о разработке собственной CRM-системы под задачи отдела ДБО.

Во второй главе была произведена разработка системы CRM. Выбраны инструменты для реализации, определены информационные потоки и построена диаграмма вариантов использования. Затем был спроектирован интерфейс внешнего вида системы, а также разработаны скрипты серверной части и скрипты фронтенд-части, спроектирована база данных.

В третьей главе представлен контрольный пример использования спроектированной CRM-системы. Последовательно выполнены шаги от авторизации в роли «администратор» до управления сделками в роли «специалист». Полученные в результате этих действий данные продемонстрированы в «журнале событий» и разделе «аналитика». Далее произведён расчёт экономической эффективности, который показал, что разработка системы CRM была обоснованной и приведёт к высвобождению дополнительных финансовых средств после срока окупаемости равного 10,7 месяцам.

Разработка данной системы позволила решить проблему автоматизации ряда задач. Приложение системы CRM работает в веб-браузере с локально установленным сервером. В дальнейшем, для улучшения функционала приложения необходимо рассмотреть разработку модуля автоматического расчёта бонусов и премий сотрудникам на основе количества проведённых сделок. Система спроектирована под нужды и задачи отдела ДБО, однако модульный подход и применение паттернов проектирования программной архитектуры кода позволит изменять или расширять необходимый функционал системы, а также задействовать сторонних разработчиков, знакомых с данными паттернами и подходами.

Теоретическая значимость данной работы заключается в систематизации и закреплении полученных в ходе обучения знаний в области моделирования бизнес-процессов организации.

Практическая значимость данной работы заключается в возможности применения разработанной CRM системы в действующем рабочем процессе отдела ДБО.

Список используемой литературы

1. АМО CRM [Электронный ресурс]. URL: <https://www.amocrm.ru> (дата обращения: 10.09.2022).
2. Битрикс 24 CRM [Электронный ресурс]. URL: www.bitrix24.ru (дата обращения: 10.09.2022).
3. Васильев, А. Программирование на PHP в примерах и задачах/ изд. Эксмо, 2021. – 352 с.
4. Веллинг, Л. Разработка веб-приложений с помощью PHP и MySQL / Л. Веллинг. - М.: Диалектика / Вильямс, 2016. - 116 с.
5. ГОСТ Р 54869 - 2011. Проектный менеджмент. Требования к управлению проектом.
6. ГОСТ Р 51904–2002. Программное обеспечение встроенных систем. Общие требования к разработке и документированию.
7. ГОССТАНДАРТ РОССИИ Руководящий документ IDEF0-2000. Методология функционального моделирования IDEF0 : введ. Постановлением Госстандарта России от 2000 г. №3.
8. Дронов В., PHP и MySQL для начинающих /СПб.: БХВ-Петербург, 2021. —432 с.: ил. — (Для начинающих)
9. Зандстра, М. PHP. Объекты, шаблоны и методики программирования/ “Диалектика-Вильямс”, 2019.- 736 с.
10. Как оценить эффективность внедрения CRM-систем: [Электронный ресурс] <https://www.sekretariat.ru/article/210383-qqe-16-m5-kak-otsenit-effektivnost-vnedreniya-crm--sistem> (дата обращения: 10.09.2022).
11. Кириченко А.В., Никольский А.П., Дубовик Е.В. Web на практике. CSS, HTML, JavaScript, MySQL, PHP для fullstack-разработчиков, 2021. — 432 с.: ил.
12. Котеров Д. В., Симдянов И. В.. PHP7 /СПб.: БХВ-Петербург, 2016. — 1088 с.: ил. — (В подлиннике)

13. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. Питер, 2019. – 816 с.
14. Сайт <https://www.mysql.com> [Электронный ресурс]. URL: <https://www.mysql.com> . (дата обращения: 10.09.2022).
15. Сайт <https://www.w3schools.com/sql/> - раздел “SQL tutorial” [Электронный ресурс]. URL: <https://www.w3schools.com/sql> (дата обращения: 10.09.2022).
16. Система управления взаимоотношениями с клиентами [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/crm> (дата обращения: 05.09.2022).
17. Скляр Д., Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов./ Издательский дом “Вильямс”, 2017 – 464с.
18. Уэнц К., PHP и MySQL. Карманный справочник / Уэнц Кристиан. - М.: Диалектика / Вильямс, 2016. - 656 с.
19. М. Фленов. PHP глазами хакера/СПб.: БХВ-Петербург, 2020. —256 с.: ил. — (4 издание)
20. Флэнаган Д., JavaScript. Полное руководство / “Диалектика-Вильямс”, 2021.- 720 с.
21. Экономические эффекты от внедрения CRM [Электронный ресурс] <https://www.cfin.ru/itm/crm/effects.shtml> (дата обращения: 10.09.2022).

Приложение А

Дамп базы данных

```
CREATE DATABASE IF NOT EXISTS `bankcrm`;
USE `bankcrm`;
CREATE TABLE IF NOT EXISTS `администраторы` (
  `id` int unsigned NOT NULL AUTO_INCREMENT,
  `Фамилия` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Имя` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Отчество` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Логин` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Пароль` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Email` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Телефон` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Номер_пк` char(50) CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci DEFAULT NULL,
  `Тип` enum('Администратор','Сотрудник','Специалист') CHARACTER
SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT
'Администратор',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT
CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

INSERT INTO `администраторы` (`id`, `Фамилия`, `Имя`, `Отчество`,
`Логин`, `Пароль`, `Email`, `Телефон`, `Номер_пк`, `Тип`) VALUES
(1, 'Макаров', 'Андрей', NULL, 'admin', 'admin', 'admin@mail.ru',
NULL, NULL, 'Администратор');

CREATE TABLE IF NOT EXISTS `журнал` (
  `id` int unsigned NOT NULL AUTO_INCREMENT,
  `id_сотрудник` int unsigned NOT NULL,
  `id_клиент` int unsigned NOT NULL,
  `Событие` enum('Зарегистрировал клиента','Изменил данные','Удалил
из системы','Принял документы','Подключил к банкингу')
CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT
NULL,
```

Продолжение Приложения А

```
`Дата` datetime DEFAULT CURRENT_TIMESTAMP,  
PRIMARY KEY (`id`),  
KEY `FK_журнал_сотрудники` (`id_сотрудник`),  
KEY `FK_журнал_клиенты` (`id_клиент`),  
CONSTRAINT `FK_журнал_клиенты` FOREIGN KEY (`id_клиент`)  
REFERENCES `клиенты` (`id`) ON DELETE CASCADE ON UPDATE  
CASCADE,  
CONSTRAINT `FK_журнал_сотрудники` FOREIGN KEY  
(`id_сотрудник`) REFERENCES `сотрудники` (`id`) ON DELETE  
CASCADE ON UPDATE CASCADE  
) ENGINE=InnoDB AUTO_INCREMENT=64 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
INSERT INTO `журнал` (`id`, `id_сотрудник`, `id_клиент`, `Событие`,  
`Дата`) VALUES  
    (58, 7, 4, 'Зарегистрировал клиента', '2022-10-04 23:08:03'),  
    (59, 7, 4, 'Изменил данные', '2022-10-04 23:08:34'),  
    (60, 7, 4, 'Удалил из системы', '2022-10-04 23:08:57'),  
    (61, 7, 5, 'Зарегистрировал клиента', '2022-10-04 23:11:49'),  
    (63, 7, 6, 'Зарегистрировал клиента', '2022-10-04 23:28:08');
```

```
CREATE TABLE IF NOT EXISTS `каналы` (  
    `id` int unsigned NOT NULL AUTO_INCREMENT,  
    `Наименование` varchar(200) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
INSERT INTO `каналы` (`id`, `Наименование`) VALUES  
    (1, 'VK'),  
    (2, 'Telegram'),  
    (3, 'Сайт'),  
    (4, 'Знакомые'),  
    (5, 'Наружная реклама'),  
    (6, 'Звонок');
```

```
CREATE TABLE IF NOT EXISTS `клиенты` (  
    `id` int unsigned NOT NULL AUTO_INCREMENT,  
    `Фамилия` varchar(200) NOT NULL,  
    `Имя` varchar(200) NOT NULL,  
    `Отчество` varchar(200) NOT NULL,  
    `Телефон` varchar(200) NOT NULL,
```

Продолжение Приложения А

```
`Канал` int unsigned NOT NULL,  
`Статус` enum('Новый','Активный','Удалён') CHARACTER SET  
utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'Новый',  
PRIMARY KEY (`id`),  
KEY `FK_клиенты_каналы` (`Канал`),  
CONSTRAINT `FK_клиенты_каналы` FOREIGN KEY (`Канал`)  
REFERENCES `каналы` (`id`) ON DELETE CASCADE ON UPDATE  
CASCADE  
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
INSERT INTO `клиенты` (`id`, `Фамилия`, `Имя`, `Отчество`,  
`Телефон`, `Канал`, `Статус`) VALUES  
  (4, 'Максимов', 'Андрей', 'Иванович', '+7987789655', 1, 'Удалён'),  
  (5, 'Иванов', 'Александр', 'Игоревич', '+7987789655', 3, 'Новый'),  
  (6, 'Борисова', 'Надежда', 'Сергеевна', '+7987789655', 5, 'Новый');
```

```
CREATE TABLE IF NOT EXISTS `сделки` (  
  `id` int unsigned NOT NULL AUTO_INCREMENT,  
  `id_сотрудник` int unsigned DEFAULT NULL,  
  `id_клиент` int unsigned DEFAULT NULL,  
  `Дата_открытия` datetime DEFAULT CURRENT_TIMESTAMP,  
  `Дата_закрытия` datetime DEFAULT NULL,  
  `Описание` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Статус` enum('Первый контакт','Подал документы','Подключён')  
CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT  
'Первый контакт',  
  PRIMARY KEY (`id`),  
  KEY `FK_накладные_персонал` (`id_сотрудник`) USING BTREE,  
  KEY `FK_сделки_клиенты` (`id_клиент`),  
  CONSTRAINT `FK_сделки_клиенты` FOREIGN KEY (`id_клиент`)  
REFERENCES `клиенты` (`id`) ON DELETE CASCADE ON UPDATE  
CASCADE,  
  CONSTRAINT `FK_сделки_сотрудники` FOREIGN KEY  
(`id_сотрудник`) REFERENCES `сотрудники` (`id`) ON DELETE  
CASCADE ON UPDATE CASCADE  
) ENGINE=InnoDB AUTO_INCREMENT=49 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
INSERT INTO `сделки` (`id`, `id_сотрудник`, `id_клиент`,  
`Дата_открытия`, `Дата_закрытия`, `Описание`, `Статус`) VALUES  
  (46, 7, 4, '2022-10-04 23:08:03', NULL, NULL, 'Первый контакт');
```

Продолжение Приложения А

(47, 7, 5, '2022-10-04 23:11:49', NULL, NULL, 'Подал документы'),
(48, 7, 6, '2022-10-04 23:28:08', NULL, NULL, 'Первый контакт');

```
CREATE TABLE IF NOT EXISTS `сотрудники` (  
  `id` int unsigned NOT NULL AUTO_INCREMENT,  
  `Фамилия` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Имя` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Отчество` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Логин` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Пароль` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Email` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Телефон` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Номер_пк` char(50) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci DEFAULT NULL,  
  `Тип` enum('Администратор','Сотрудник','Специалист') CHARACTER  
SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT 'Сотрудник',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;  
  
INSERT INTO `сотрудники` (`id`, `Фамилия`, `Имя`, `Отчество`,  
  `Логин`, `Пароль`, `Email`, `Телефон`, `Номер_пк`, `Тип`) VALUES  
  (7, 'Иванов', 'Олег', 'Николаевич', 'fast1358', 'fast1358', NULL,  
  NULL, NULL, 'Специалист');
```

Приложение Б
Листинг функции dbGo()

```
function dbGo()
{
try {
$db = new PDO('mysql:host=' . DB_HOST . ';dbname=' . DB_NAME, DB_USER,
DB_PASS);
$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
$db->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE,
PDO::FETCH_ASSOC);
} catch (PDOException $e) {
print "Ошибка подключения: " . $e->getMessage() . "<br/>";
die();
}
return $db;
}
```

Приложение В
Листинг функции sql()

```
function sql($q): PDOStatement
{
try {
$n = dbGo()->query($q);
} catch (PDOException $e) {
print "Ошибка : " . $e->getMessage() . "<br/>";
die();
}
return $n;
}
```

Приложение Г

Листинг функции sqlInsert()

```
function sqlInsert($table, $arrValues = []): int
{
    $keys = array_keys($arrValues);
    $values = array_values($arrValues);
    $valuesPrepared = array_map(function ($item) {
        $type = gettype($item);
        if ($type === 'double' or $type === 'integer') {
            return $item;
        }
        return "" . (string)$item . "";
    }, $values);
    $keyesPrepared = array_map(function ($item) {
        return "" . (string)$item . "`";
    }, $keys);
    $strKeys = implode(", ", $keyesPrepared);
    $strValues = implode(", ", $valuesPrepared);
    $query = "INSERT INTO $table ($strKeys) VALUES ($strValues)";
    if(empty($arrValues)){
        $query = "INSERT INTO $table VALUES()";
    }
    return dbGo()->exec($query);
}
```

Приложение Д

Листинг функции `template()`

```
function template(string $view, array $arr = [], string $path = './templates/'): string
{
    extract($arr);
    ob_start();
    include($path . $view . '.php');
    return ob_get_clean();
}
```