|  |  |  |  |
| --- | --- | --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН  СЕВЕРО-КАЗАХСТАНСКИЙ УНИВЕРСИТЕТ ИМ. М. КОЗЫБАЕВА | | | |
|  | | | |
|  | | |  |
| СЕМЕСТРОВЫЙ ПРОЕКТ  по дисциплине «Инструментальные средства разработки программ» | | | |
| Выполнили студенты  группы АПО-20-1 |  | Зайкина Л.И.  Валиев Д.Т. | |
| Руководитель,  доцент кафедры ИКТ |  | Астапенко Н.В. | |

Петропавловск, 2023

СОДЕРЖАНИЕ

[ПЕРЕЧЕНЬ ИСПОЛЬЗУЕМЫХ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ 3](#_Toc133885553)

[ВВЕДЕНИЕ 4](#_Toc133885554)

[1 Аналитическая часть 6](#_Toc133885555)

[1.1 Характеристика/описание объекта исследования 6](#_Toc133885556)

[1.2 Обзор аналогичных проектов 7](#_Toc133885557)

[1.3 Техническое задание 9](#_Toc133885558)

[1.4 Функциональное моделирование (IDEF0, IDEF3). 11](#_Toc133885559)

[2 Проектная часть 17](#_Toc133885560)

[2.1 Проектирование информационных потоков (DFD) 17](#_Toc133885561)

[2.2 Проектирование средствами UML 18](#_Toc133885562)

[2.3 Проектирование баз данных 25](#_Toc133885563)

[2.4 Проектирование пользовательских интерфейсов 27](#_Toc133885564)

[ЗАКЛЮЧЕНИЕ 39](#_Toc133885565)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 40](#_Toc133885566)

ПЕРЕЧЕНЬ ИСПОЛЬЗУЕМЫХ ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

В отчете по семестровому проекту имеются следующие обозначения и сокращения:

|  |  |  |
| --- | --- | --- |
| VBA | – | программный язык, разработанный для автоматизации задач и создания макросов в приложениях Microsoft Office и других программных продуктах.; |
| DFD | – | диаграмма, которая отображает потоки данных и процессы, в которых они используются в системе. Она используется для моделирования бизнес-процессов и описания того, как данные передаются между различными частями системы; |
| IDEF0 | – | методология моделирования бизнес-процессов, использующая IDEF (Integrated Definition for Function Modeling) диаграммы, которые отображают структуру функциональных блоков, их связи и иерархию; |
| IDEF3 | – | методология моделирования и стандарт документирования процессов, происходящих в системе; |
| UML | – | стандартный графический язык моделирования, используемый для визуального описания, проектирования и документирования системного архитектурного проектирования; |
| ERD | – | стандартный графический язык моделирования, используемый для визуального описания, проектирования и документирования системного архитектурного проектирования; |
| КТ | – | коэффициент технического обустройства; |
| МАП | – | месячная арендная плата; |
| ИИН | – | индивидуальный идентификационный номер физического лица; |
| БИН | – | бизнес-идентификационный номер юридического лица; |
| БД | – | база данных; |
| РК | – | Республика Казахстан. |

# ВВЕДЕНИЕ

В настоящее время автоматизация бизнес-процессов является одним из основных трендов в информационных технологиях. Разработка необходимой для организации информационной системы позволяет повысить эффективность работы предприятия, сократить время на обработку данных и уменьшить риски возможных ошибок. Кроме того, автоматизация бизнес-процессов является необходимым шагом для тех компаний, которые стремятся к цифровой трансформации и повышению конкурентоспособности на рынке. Информационные системы позволяют не только автоматизировать рутинные задачи, но и повышать уровень управления организацией, а также ускорять процессы принятия решений на основе данных. Это особенно важно для компаний, которые занимаются арендой нежилых помещений, где точность расчетов и своевременность обработки данных играют важную роль.

Организации предоставляющие услуги по аренде нежилых помещений становятся популярнее с каждым годом, так как в современном мире возрастает количество малых бизнесов, которым необходимы здания для их деятельности. Отсюда следует, что одним из важных элементов в работе агентства недвижимости является учет арендной платы за нежилые помещения. Для компаний важно вести учет для аналитики и отслеживания корректности работы. Данный процесс может быть упрощен и оптимизирован с помощью автоматизации бизнеса.

В рамках данного семестрового проекта рассматривается разработка информационной системы для учета арендной платы за нежилые помещения «Альфа», используя инструментальную среду разработки VBA, язык программирования, используемый для создания макросов и приложений в Microsoft Office. Важным аспектом разработки информационной системы на VBA является то, что данная среда разработки широко используется в сфере бизнеса и финансов, что позволяет быстро интегрировать разработанную систему в уже существующие процессы. Одним из основных преимуществ разработки информационной системы на VBA является возможность автоматического расчета арендной платы за нежилые помещения на основе заранее заданных параметров. Таким образом, пользователь может быть уверен в том, что расчеты проводятся корректно и точно. В разработанной информационной системе также будет реализована возможность формирования отчетов и аналитических данных по оплате аренды, что позволит агентству недвижимости контролировать финансовые потоки и принимать взвешенные решения на основе данных. Кроме того, система будет обеспечивать удобный интерфейс для пользователей, что позволит быстро и эффективно работать с информацией. Кроме того, VBA имеет широкую функциональность, что дает возможность реализовать различные сценарии работы с данными и создать настраиваемые интерфейсы для пользователей. Использование данного функционала позволит автоматизировать рутинные задачи компании.

Актуальность разработки информационной системы «Альфа» на VBA обусловлена необходимостью оптимизации учета арендной платы, что является одной из ключевых задач бизнеса. Ручной учет этой информации занимает много времени и сопряжен с риском ошибок. Автоматизация процесса учета поможет не только уменьшить риски возможных ошибок, но и ускорить обработку данных, а также повысить точность расчета арендной платы.

Объектом исследования данной курсовой работы является процесс автоматизации учета арендной платы за нежилые помещения в компаниях, занимающихся арендой недвижимости.

Предметом исследования является разработка информационной системы на VBA для учета арендной платы, которая будет позволять автоматически рассчитывать арендную плату на основе заранее заданных параметров, формировать отчеты и аналитические данные по аренде, а также обеспечивать удобный интерфейс для пользователей.

Целью данной курсовой работы является разработка информационной системы на VBA для учета арендной платы за нежилые помещения.

Для достижения указанной цели были поставлены следующие задачи:

* проанализировать предметную область;
* спроектировать разрабатываемую информационную систему посредством разработки диаграмм IDEF0, IDEF3, DFD, UML, ERD;
* реализовать базу данных, хранящуюся в таблицах MS Excel;
* реализовать интерфейсы для выполнения предполагаемого функционала программы.

В рамках данного проекта будут использоваться следующие методы исследования:

* анализ требований пользователя: проведение опроса пользователей и анализ требований, чтобы определить функциональные и нефункциональные требования к информационной системе;
* исследование существующих систем: проведение анализа уже существующих систем учета арендной платы и анализ их функциональности, преимуществ и недостатков.

Практическая значимость работы заключается в возможности использования разработанной информационной системы для автоматизации учета арендной платы за нежилые помещения в любой организации. Результаты данной работы помогут оптимизировать бизнес-процессы, ускорить обработку данных и повысить точность расчетов арендной платы.

В целом, разработка информационной системы на VBA для учета арендной платы за нежилые помещения является актуальной и практически значимой задачей, которая поможет ускорить процессы работы с данными, сократить риски возможных ошибок и повысить эффективность работы компаний, занимающихся арендой помещений.

# 1 Аналитическая часть

## 1.1 Характеристика/описание объекта исследования

В рамках семестрового проекта необходимо разработать приложения на VBA Excel для учета арендной платы компании за нежилые помещения. Приложение имеет простой и интуитивно понятный интерфейс, который позволяет пользователям быстро и легко вносить и отслеживать данные об аренде.

Программа для учета арендной платы на VBA Excel будет актуальна для компаний, которые сдают в аренду нежилые помещения. Такие компании смогут использовать приложение для автоматизации процессов управления арендой, отслеживания выплат арендной платы и управления договорами об аренде. Приложение будет полезно для финансового отдела компании, который сможет легко отслеживать сроки оплаты и сроки действия договоров. Таким образом, программа позволит оптимизировать финансовые процессы и уменьшить риски, связанные с несвоевременными или неправильными выплатами. Также, приложение может быть использовано в качестве инструмента для управления недвижимостью, таким образом, оно может быть актуально для владельцев недвижимости, управляющих компаний и агентов по недвижимости. Они могут использовать приложение для отслеживания договоров об аренде, учета арендных платежей и для генерации отчетности по недвижимости.

Программа имеет следующие особенности:

* справочник арендаторов. Необходим для хранения информации об арендаторах нашей компании. Данный справочник делится на физические и юридические лица. Для физического лица заполняется следующее: ФИО, паспортные данные, ИНН, адрес, контактный телефон. Для юридического лица: Содержит наименования арендатора, юридический адрес арендатора, ФИО руководителя предприятия арендатора, контактный телефон арендатора, ИНН, номер и дата лицензии о деятельности. Имеется возможность добавления новой записи, удаления и редактирования существующей;
* справочник объектов (помещений) аренды. Необходим для хранения информации о помещениях, которые организация сдает в аренду. Данный справочник содержит код объекта аренды, название, адрес, площадь (кв.м), площадь подвала (при наличии), коэффициент подвала (значение от 0 до 1), коэффициент технического обустройства помещения (КТ) (значение от 1 до 2). Имеется возможность добавления новой записи, удаления и редактирования существующей;
* справочник договоров об аренде. Необходим для учета договоров, оформляемых компанией. Также подразделяется на две категории: для юридический и для физических лиц. Данный справочник содержит номер договора, дату начала аренды, дату окончания аренды, срок аренды, дату заключения договора, арендатора, помещение, ежемесячная дату оплаты аренды и месячный размер оплаты. Имеется возможность добавления нового договора, удаления и редактирования существующего;
* при создании записи договора производится автоматическое заполнение шаблона договора данными из программы в Microsoft Word, это позволит сократить время, так как больше не будет ручного заполнения договора;
* программа позволяет формировать отчеты для аналитики работы компании, такие как отчет об итоговой сумме прибыли за выбранный месяц, список действующих арендаторов на выбранную дату, список помещений, несданных в аренду на выбранную дату.
* программа подразумевает разделение функционала для администратора и менеджера, администратор имеет возможность добавлять новых пользователей компании, а также управлять справочником пользователей.

Таким образом, приложение «Альфа» на VBA Excel позволяет пользователям управлять арендной платой компании, вносить и отслеживать данные об арендных платежах, получать уведомления об оплате аренды и упрощать процесс управления арендной платой для компании. То есть в целом, приложение может быть полезным для любой компании, занимающейся арендой нежилых помещений, которая хочет оптимизировать свои процессы управления арендой, уменьшить риски, связанные с несвоевременными выплатами арендной платы и улучшить аналитическую отчетность.

## Обзор аналогичных проектов

Были проанализированы аналогичные проекты по данной тематике:

* C2Soft – это программа, которая содержит полный комплекс модулей, связанных c управлением, арендой и продажей недвижимости. Использование системы позволяет проводить финансовый анализ. Автоматизация учета правоудостоверяющей, технической и страховой документации по объектам недвижимого имущества. Одним из минусов C2Soft является высокая стоимость, что может быть недоступно для малых и средних компаний. Кроме того, программа может оказаться слишком сложной для использования для некоторых пользователей;
* Yardi Voyager – эта программа используется для учета аренды коммерческой недвижимости. Она предоставляет широкий функционал, включая управление договорами, финансовый учет и отчетность. Однако, Yardi Voyager может быть дорогой для малых и средних компаний;
* LeaseAccelerator – это программное обеспечение для управления лизинговыми договорами, которое предоставляет аналитические инструменты для управления арендой и оптимизации расходов на аренду. LeaseAccelerator является интегрированной платформой, которая позволяет автоматизировать управление лизингами в различных отраслях. Главный минусом LeaseAccelerator является то, что программа может быть слишком сложной для использования для некоторых пользователей;
* RentMaster – программа для учета арендной платы, которая позволяет хранить информацию о договорах аренды, арендаторах и объектах аренды. Основным недостатком RentMaster является отсутствие функционала автоматического заполнения договоров и возможности формировать отчеты для аналитики работы компании;
* Rental Property Manager – программа для управления арендой недвижимости, которая помогает в учете арендной платы, оплате расходов на содержание объектов недвижимости и контроле договоров. Rental Property Manager также имеет функции отчетности, которые помогают в анализе прибыли и потерь. Недостаток Rental Property Manager в том, что программа может быть недостаточно гибкой для некоторых компаний, требующих индивидуальных настроек;
* Propertyware – программа для управления арендой недвижимости, которая включает в себя учет арендной платы, управление договорами, учет расходов и отчетность. Propertyware также имеет функции автоматизации процессов и интеграции с другими системами, такими как Zillow и Apartments.com. Минусом программы является высокая стоимость и сложность в настройке для конкретной компании;
* Buildium – программа для управления арендой недвижимости, которая позволяет управлять арендой как жилых, так и коммерческих помещений, включая учет арендной платы, управление договорами и отчетность. Buildium также имеет функции управления собственностью, которые помогают в учете расходов и контроле бюджета. Недостатком программы является то, что некоторые пользователи могут считать пользовательский интерфейс не интуитивным и сложным для использования;
* Арендодатель PRO – это программное обеспечение позволяет вести учет арендных платежей, расчет комиссионных и налогов, а также автоматически формировать отчетность для налоговых органов. Кроме того, в программе есть функции для учета арендаторов и объектов недвижимости, а также для планирования будущих доходов от арендных платежей. Одним из минусов программы "Арендодатель PRO" может являться ее сложность в использовании для непрофессионалов. Некоторые функции могут потребовать определенных знаний в области бухгалтерского учета и налогообложения, что может создавать трудности для новых пользователей. Также, стоит учитывать, что программа "Арендодатель PRO" разработана исключительно для российского рынка и может не подходить для использования в других странах с различными законодательством и налоговыми требованиями.

Каждое из этих программных приложений имеет свои преимущества и недостатки в зависимости от потребностей и предпочтений пользователя в учете арендной платы. Некоторые из них предоставляют больше функций для управления лизинговыми договорами и оптимизации расходов, а другие – больше возможностей для учета правоудостоверяющей, технической и страховой документации. После исследования аналогов был разработан концепт, согласно которому приложение должно быть максимально простым в использовании и содержать функционал для учета арендной платы, контроля договоров и отчетности, чтобы охватить больший сегмент пользователей, включая малые и средние компании.

## Техническое задание

Тип информационной системы: приложение на VBA Excel для учета арендной платы компании за нежилые помещения.

Назначение и цель системы: предоставить возможность пользователям управлять арендной платой компании, вносить и отслеживать данные об арендных договорах, формировать печатные формы договоров, получать аналитические данные и упрощать процесс управления арендной платой для компании.

Характеристики объектов автоматизации:

К объектам автоматизации системы относятся:

* справочник объектов;
* справочник арендаторов;
* справочник договоров;
* отчеты.

Требования к системе:

Система должна соответствовать следующим требованиям:

* интерфейс системы должен быть понятен для пользователей;
* система должна быть удобно управляемой и настраиваемой;
* система должна гарантировать безопасность;
* система должна позволять создавать, изменять и удалять договора, арендаторов и объекты;
* система должна обрабатывать большие объемы данных.

Система должна содержать:

* средства хранения информации;
* средства отбора информации по различным критериям.

Требования к архитектуре системы:

* в качестве основного языка программирования используется VBA (Visual Basic for Applications);
* интерфейс пользователя реализуется с помощью пользовательских форм;
* данные хранятся в листах Excel.

Требования к аппаратному обеспечению:

* компьютер с операционной системой Windows;
* версия Microsoft Excel, поддерживающая VBA.

Требования к сети:

* система не требует подключения к сети.

Дизайн интерфейса:

* интерфейс приложения должен быть интуитивно понятным и простым для пользователя;
* кнопки и элементы управления должны быть наглядными и легко обозримыми;
* цветовая гамма приложения должна быть спокойной и не вызывать раздражения у пользователя.

Схема базы данных:

* база данных представлена в виде листов Excel;
* каждый договор, арендатор и объект представлен отдельной строкой в соответствующей таблице, с указанием его дополнительных критериев;
* данные для входа каждого пользователя сохраняются в соответствующей таблице.

Требования к режимам функционирования:

* разрабатываемая система должна функционировать 24 часа в сутки, 365 дней в году, в зависимости от режима работы организации.

Пользователи:

* Система должна иметь два типа пользователей: администраторы и менеджеры.

Функционал, выполняемый пользователями:

* менеджер может просматривать список арендаторов и их контактные данные, а также просматривать информацию о доступных помещениях. Он также может вносить изменения в данные справочники или удалять данные. Менеджер может заключать договоры об аренде с физическими и юридическими лицами, формировать печатные формы и осуществлять редактирование или удаление в данных справочниках. Менеджер может также формировать отчеты для аналитики работы компании, такие как отчет об итоговой сумме прибыли за выбранный месяц, список действующих арендаторов на выбранную дату, список помещений, несданных в аренду на выбранную дату;
* администратор имеет возможность добавлять новых пользователей компании, а также управлять справочником пользователей. Он также может добавлять, редактировать и удалять записи в справочниках арендаторов и объектов аренды. Администратор может создавать новые договоры об аренде и редактировать существующие. Также он может формировать отчеты, аналогичные менеджеру.

Функционал, выполняемый системой:

* расчет месячной арендной платы на основе данных о помещение;
* разделение функционала в зависимости от должности пользователя.

Состав и содержание работ по созданию системы:

* разработать модель базы данных для хранения и обработки всех необходимых данных для учета арендной платы на VBA. База данных должна иметь возможность хранить информацию о договорах, арендатор, объектах и пользователях;
* разработать удобный пользовательский интерфейс, который позволит пользователям легко добавлять, удалять, редактировать и просматривать необходимые данные;
* реализовать возможность создания печатной формы договора в формате Word-документа.

Порядок контроля и приемки системы.

Для проверки выполнения указанных функций системы, определения и проверки соответствия требованиям ТЗ количественных и качественных характеристик системы, выявления и устранения недостатков в действиях системы и в разрабатываемой документации, поэтапный контроль за ходом разработки, должны быть проведены следующие виды тестирований:

1. предварительное тестирование: проводится на стадии разработки для выявления дефектов и обеспечения соответствия системы первоначальным требованиям. Данный этап включает модульное тестирование, интеграционное тестирование и системное тестирование;
2. приемочное тестирование: проводится после разработки для проверки соответствия системы критериям приемки и достижения целей, указанных в требованиях. Этот этап может быть разделен на альфа- и бета-тестирование, которые проводятся конечными пользователями и ограниченным числом внешних тестировщиков соответственно;
3. тестирование конечных пользователей: проводится для проверки соответствия системы потребностям и ожиданиям конечных пользователей. Этот этап включает тестирование системы конечными пользователями и предоставление отзывов команде разработчиков [3].

## Функциональное моделирование (IDEF0, IDEF3).

IDEF0 – это методология и нотация для функционального моделирования системы, предназначенная для формализации и описания бизнес-процессов, поддерживаемых и автоматизируемых этой системой.

Диаграммы этой методологии представляют логические отношения между работами – функциональными блоками, если говорить в терминах методологии, – в разрезе «вход-выход-условие-ограничение». Этот подход позволяет описать систему на любом желаемом уровне детализации функционала и сформировать общее представление о её назначении[4].

Контекстная диаграмма IDEF0 для веб-приложения онлайн библиотеки – это графическое представление основных функций и взаимодействий системы на высоком уровне абстракции. Целью диаграммы является показать основные функции системы и ее взаимодействие с внешним миром. Диаграмма может быть использована для выявления основных проблем, связанных с проектированием системы, и для определения потребностей пользователей и других заинтересованных сторон.

Контекстная диаграмма представлена в соответствии с рисунком 1.1.

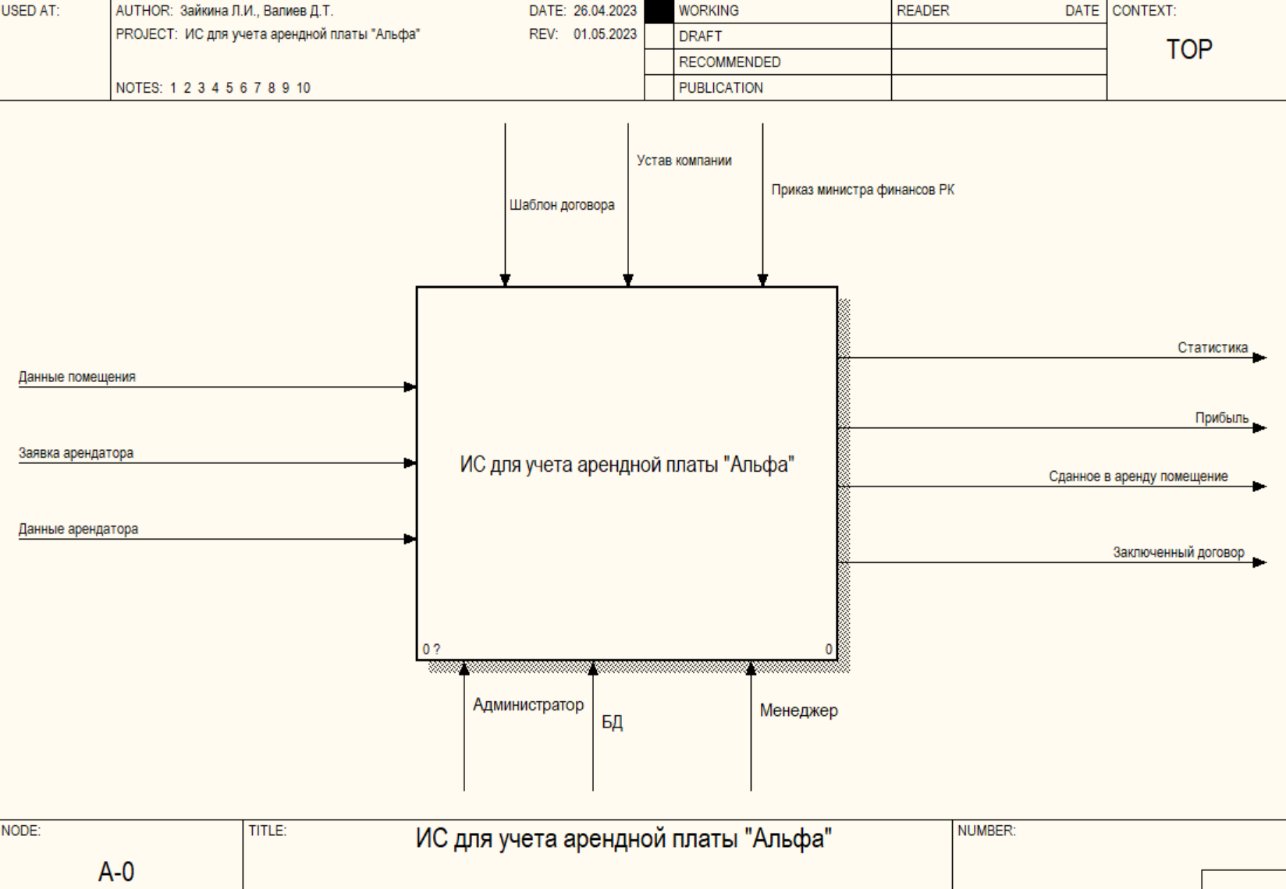


Рисунок 1.1 Контекстная диаграмма

Контекстная диаграмма изображает деятельность самого верхнего уровня и обозначает границу моделирования относительно цели, возможностей и точки зрения. На контекстной диаграмме специфицируемая система представляется в виде одного единственного процесса, связанного с внешними сущностями потоками данных. Контекстная диаграмма представляет требования к системе на самом верхнем уровне – уровне взаимодействия с окружением.

На контекстной диаграмме изображен процесс ИС для учета арендной платы «Альфа», его входные и выходные данные, а также механизмы и управление. К входным данным относятся «Данные помещения» и «Заявка арендатора», «Данные арендатора», «Данные арендатора». К выходным данным относятся: «Статистика», «Прибыль» и «Сданное в аренду помещение», «Заключенный договор». Механизмами являются «База данных», «Администратор» и «Менеджер». Управлением здесь являются «Шаблон договора», «Устав компании», «Приказ министра финансов РК».

Процесс работы ИС для учета арендной платы «Альфа» включает в себя 3 процесса: «Управление арендой», «Создание договора аренды», «Управление данными аренды»

Заявка арендатора, данные арендатора, данные помещения приходят на вход в процесс «Управление арендой», после действий принятия и обработки данных происходит формирование решения о сдаче в аренду запроса клиента и производится создание договора аренды. Формируется шаблон договора, заполняется обработанными данными, после чего производится заключение договора и на выходе получается заключенный договор. Далее исходя из информации по заключению договора (данные по сдаче в аренду и данные о платежах) производится формирование статистики, подсчет прибыли, вывод помещений, сданных в аренду.

Декомпозиция используется при моделировании информационных систем для разделения функций на составляющие части. Диаграмма декомпозиции процесса ИС для учета арендной платы «Альфа» представлена в соответствии с рисунком 1.2.

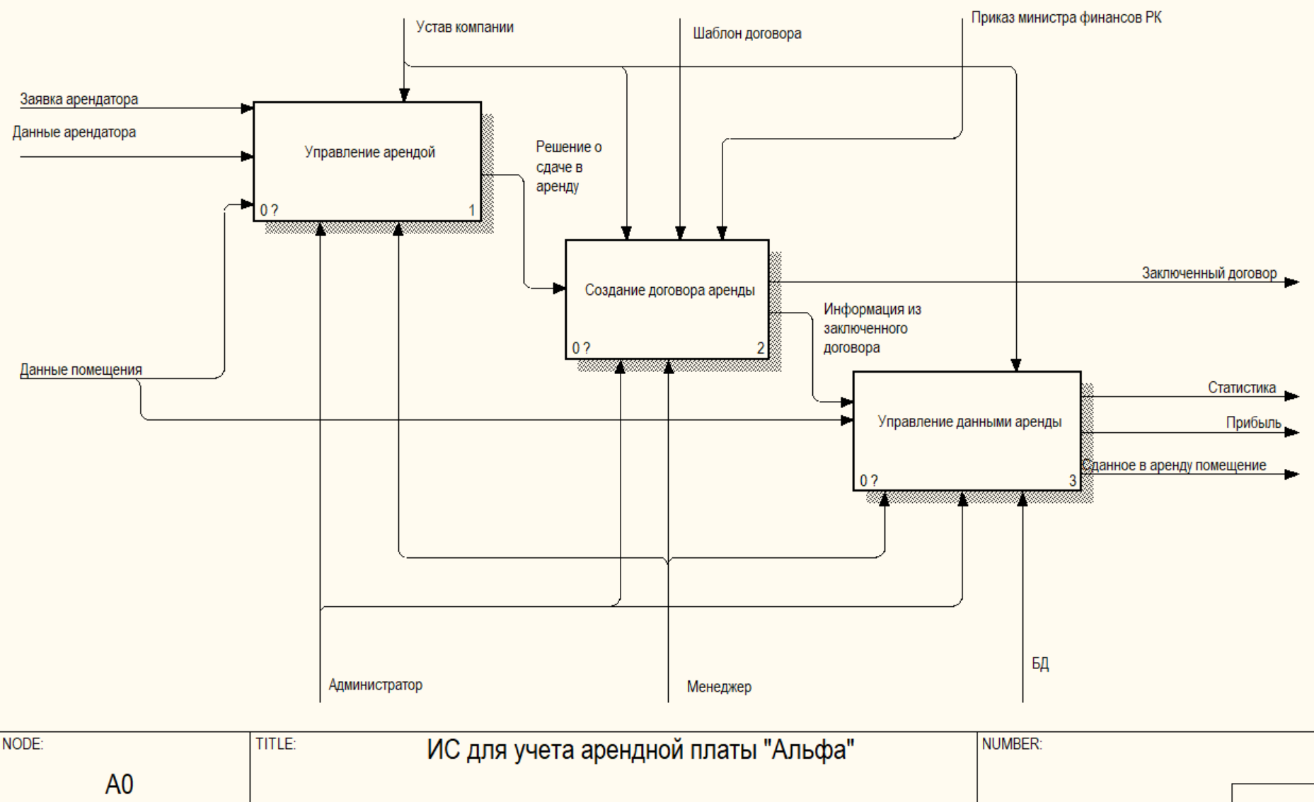


Рисунок 1.2 Диаграмма декомпозиции IDEF0

Диаграммы декомпозиции предназначены для детализации функций и получаются при разбиении контекстной диаграммы на крупные подсистемы (функциональная декомпозиция) и описывающие каждый подсистему и их взаимодействие. Единственная функция, представленная на контекстной диаграмме верхнего уровня, может быть разложена на основные подфункции посредством создания дочерней диаграммы [4, 5].

Создание диаграмм IDEF0 является важным шагом в проектировании любой системы, в том числе и онлайн библиотеки. Они помогают улучшить понимание системы, определить ее ключевые компоненты и функциональность, а также ускорить процесс разработки и тестирования системы.

IDEF3 (Integration Definition for Process Description Capture Method) - это методика захвата и описания процессов, которая развивает и расширяет идеи, представленные в IDEF0. IDEF3 обеспечивает формальный способ моделирования информационных потоков и их взаимодействия в процессах.

В рамках IDEF3 процессы рассматриваются как совокупность входящих и исходящих потоков информации. Методика позволяет описывать различные аспекты процесса, включая функции, связи между функциями и потоки информации, которые передаются между функциями.

IDEF3 использует специальный графический язык для представления процессов, включая блоки, стрелки и атрибуты. Блоки представляют функции или роли, которые выполняются в рамках процесса, а стрелки обозначают потоки информации между функциями. Атрибуты могут использоваться для описания характеристик функций или потоков данных.

IDEF3 также может использоваться в качестве коммуникационного средства для обмена информацией между участниками проекта или для документирования текущих процессов. Методика обеспечивает структурированный и формализованный подход к моделированию процессов, что упрощает понимание и анализ сложных систем.

Далее проведен второй уровень декомпозиции в нотации IDEF3.

Диаграмма декомпозиции блока «Управление арендой» представлена в соответствии с рисунком 1.3.

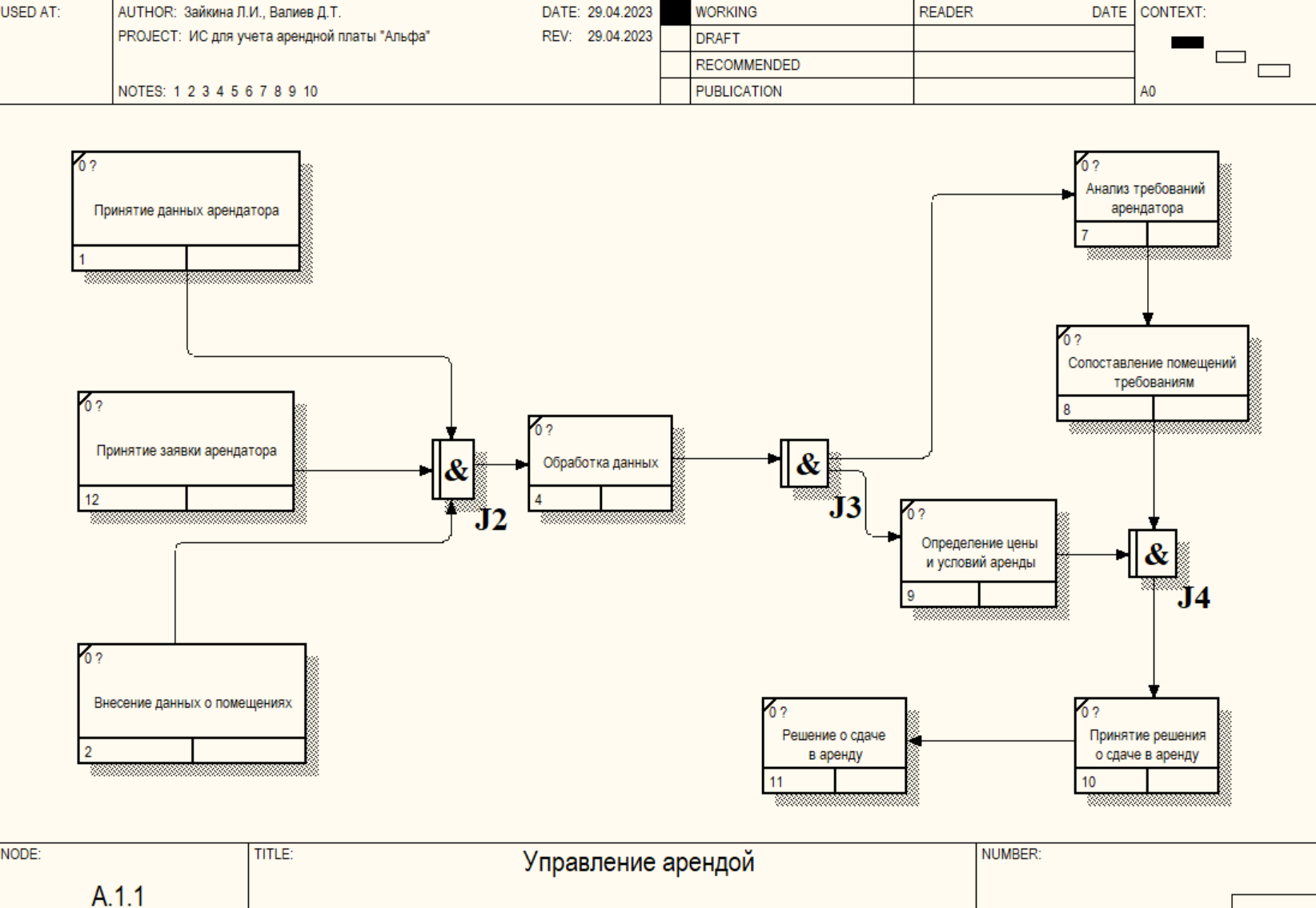


Рисунок 1.3 Диаграмма декомпозиции IDEF3

На данной диаграмме декомпозиции указаны действия по принятию данных арендатора, заявки арендатора, а также внесение данных о помещениях. Далее данные обрабатываются, и осуществляется следующие этапы, включающие в себя анализ требований арендатора, сопоставление помещений требованиям, определение цены и условий аренды. После данных этапов происходит действие принятия решения о сдаче в аренду, на выходе получается решение о сдаче в аренду.

Диаграмма декомпозиции блока «Создание договора аренды» представлена в соответствии с рисунком 1.4.

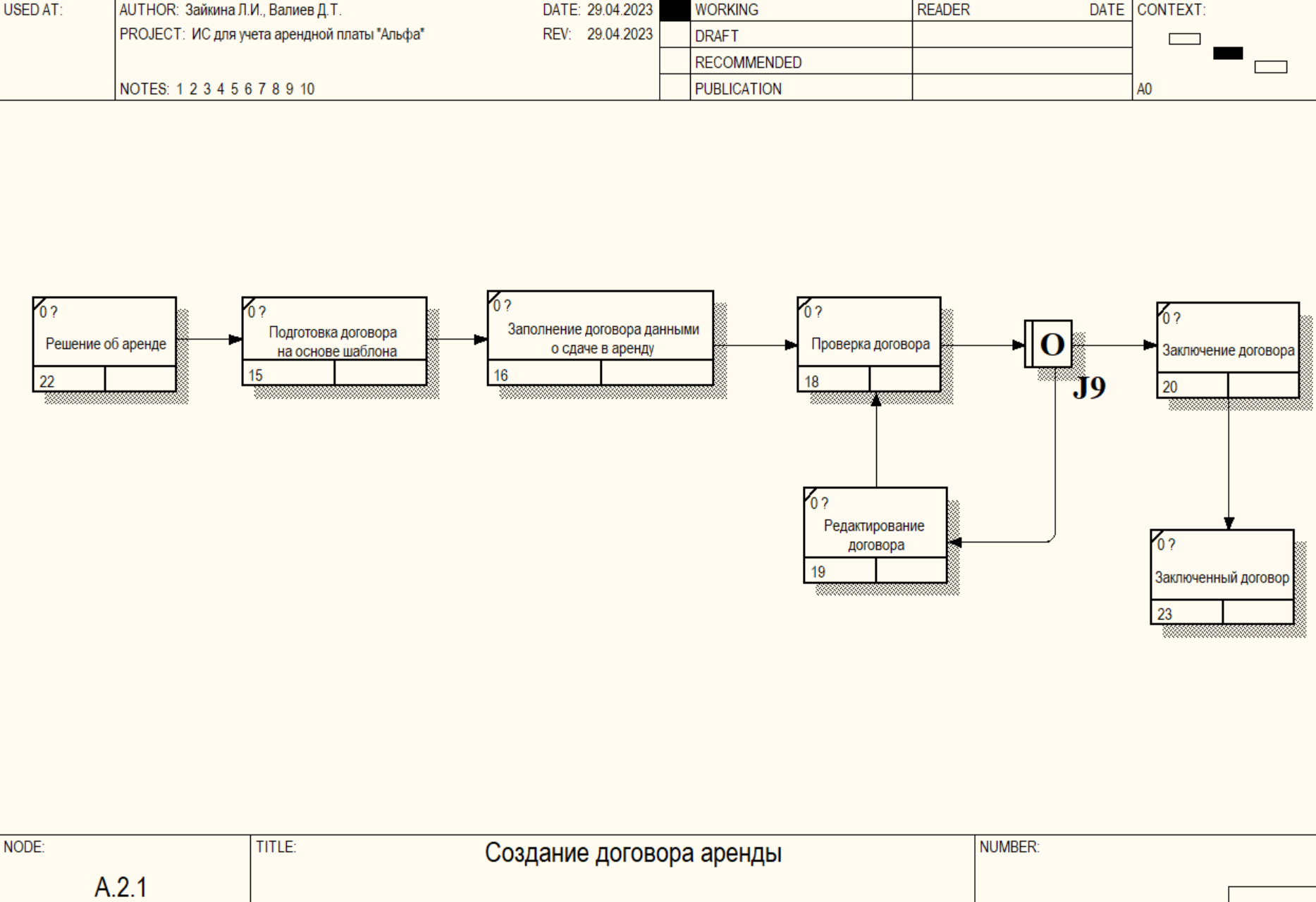


Рисунок 1.4 Диаграмма декомпозиции IDEF3

На данной диаграмме декомпозиции на вход подается решение об аренде. По решению об аренде происходит заполнение договора по шаблону. Далее договор проверяется, и при необходимости существует возможность его редактирования. По окончании заполнения и проверки договора происходит его заключение путем подписания двумя сторонами, и на выходе получается заключенный договор с условиями и обязательствами по аренде.

На диаграмме декомпозиции блока «Управление данными» на вход подается информация из заключенного договора, а именно данные о сдаче в аренду, данные о платежах. Далее идет блок выбора действий, в котором по выбору есть возможность формирования статистики, подсчет прибыли, вывод договора на экран, а также существует возможность его печати, а также вывод списка помещений.

Диаграмма декомпозиции блока «Управление данными» представлена в соответствии с рисунком 1.5.

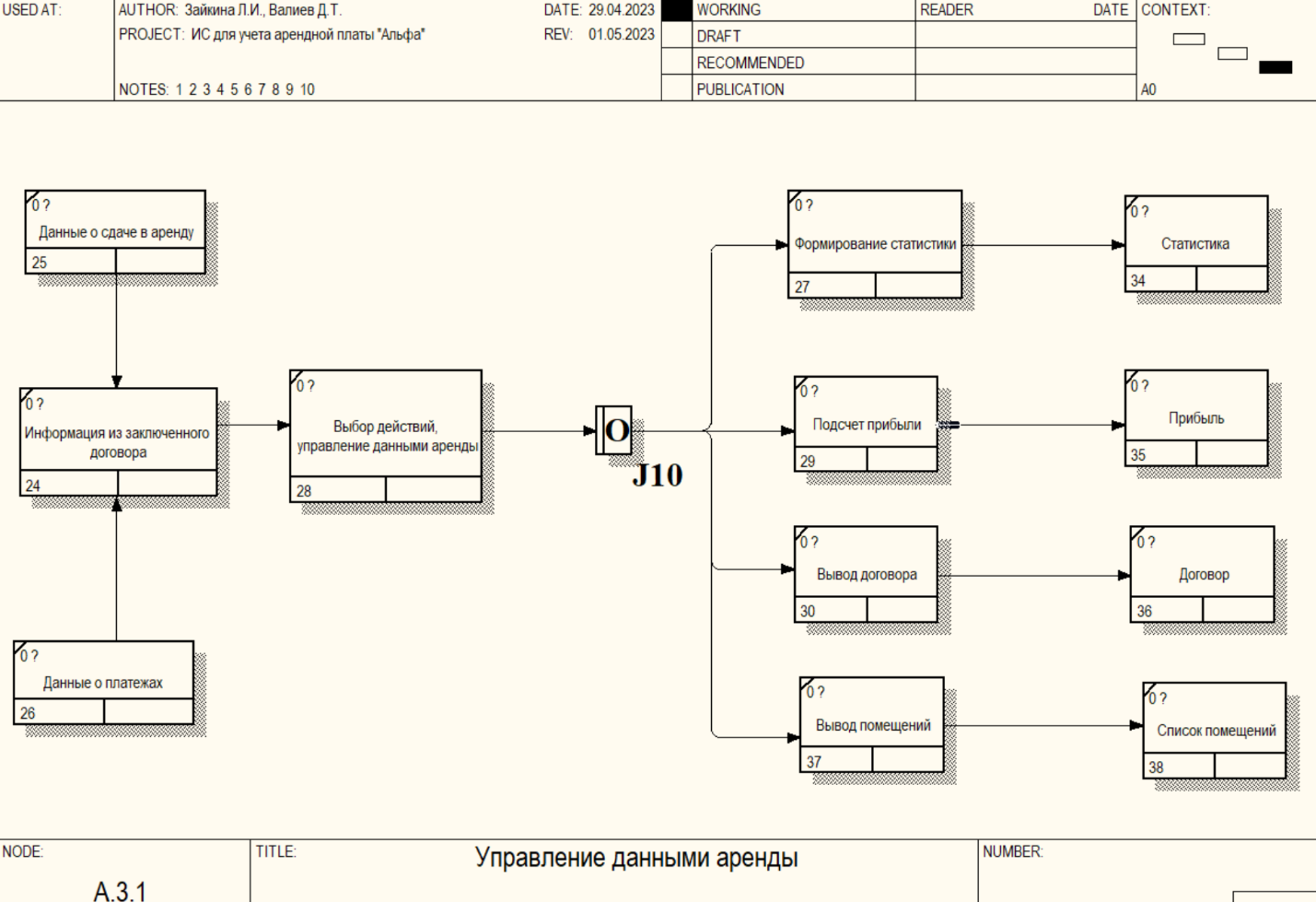


Рисунок 1.5 Диаграмма декомпозиции IDEF3

В целом, IDEF3 является мощным инструментом для моделирования процессов, который помогает описать информационные потоки и их взаимодействие в системе. Он способствует улучшению процессов, оптимизации потоков информации и обмену информацией между участниками проекта.

# 2 Проектная часть

2.1 Проектирование информационных потоков (DFD)

DFD (Data Flow Diagram) - это графическая модель, используемая для представления потоков данных в системе, где акцент делается на данных, а не на активных лицах или процессах. Она состоит из блоков, называемых процессами, которые представляют действия, осуществляемые над данными. Также в DFD присутствуют стрелки, которые показывают потоки данных между процессами, внешними сущностями и хранилищами данных.

Главное преимущество DFD состоит в том, что он позволяет выявить потоки данных между процессами и определить, какие данные передаются и обрабатываются внутри системы. Это помогает анализировать и оптимизировать потоки данных, выявлять узкие места и возможности для улучшений в системе. DFD также помогает в идентификации связей между различными компонентами системы и понимании, как данные перемещаются через систему [].

Контекстная диаграмма указана в соответствии с рисунком 2.1.

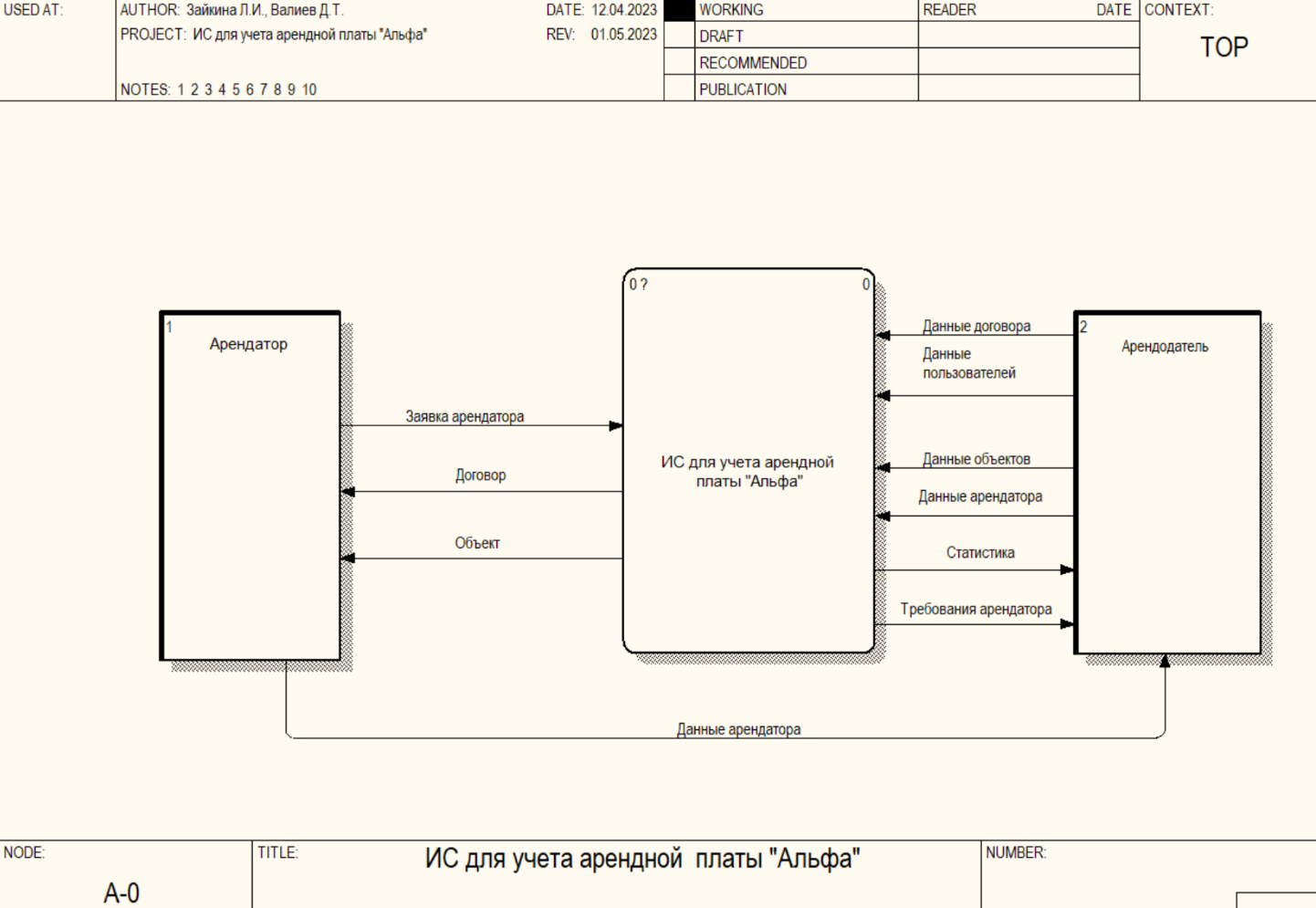


Рисунок 2.1 Контекстная диаграмма

На контекстной диаграмме сущности «Арендатор» и «Арендодатель», указаны потоки информации от клиента и арендодателя в лице менеджера и администратора в информационную систему, причем данные арендатора в первую очередь передаются напрямую арендодателю, затем арендодатель обрабатывает данные арендатора и вносит их в информационную систему.

Диаграмма декомпозиции контекстной диаграммы указана в соответствии с рисунком 2.2.

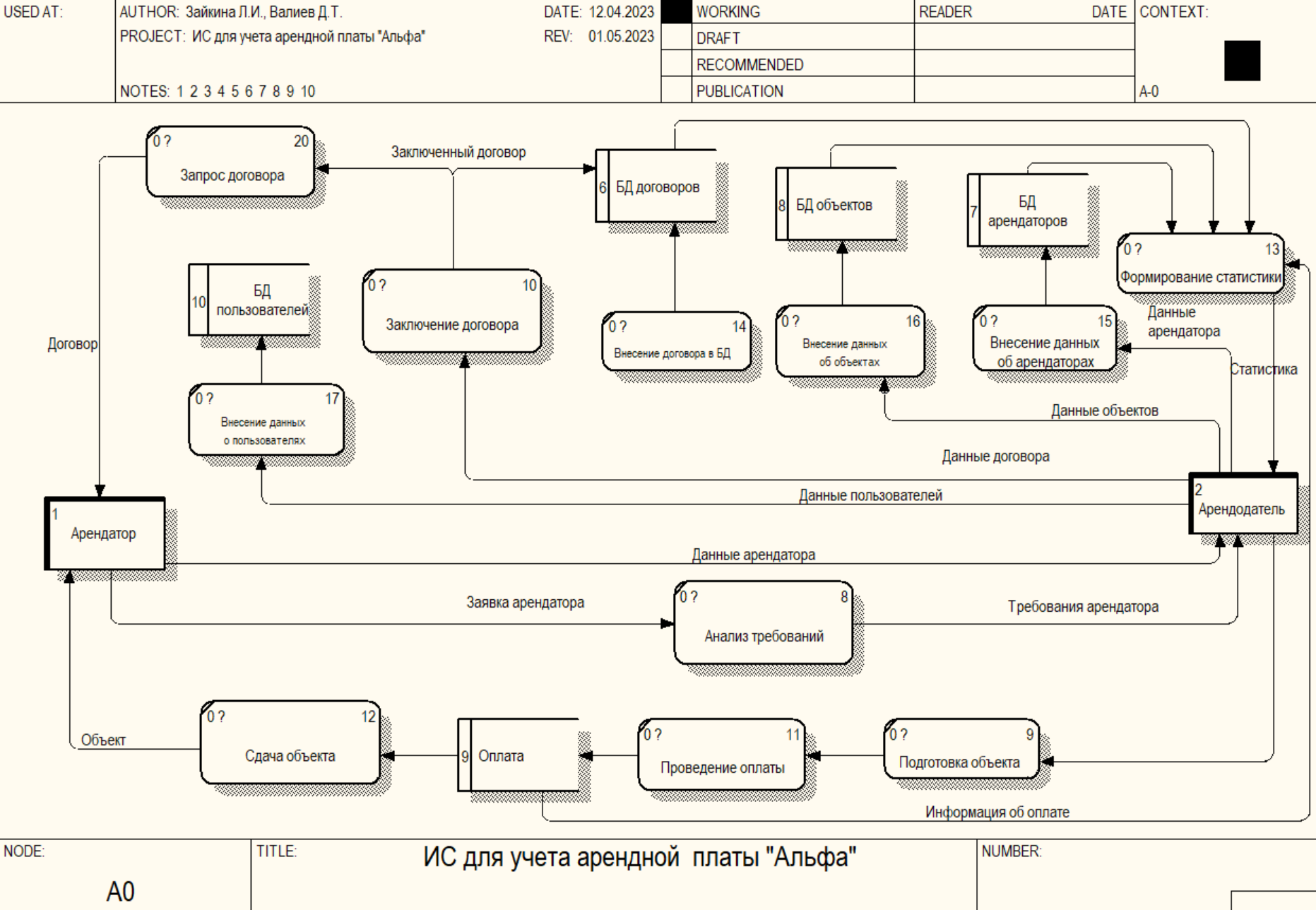


Рисунок 2.2 Декомпозиция контекстной диаграммы

На диаграмме декомпозиции указаны потоки информации, процессы, совершающиеся в информационной системе, сущности, и хранилища данных.

DFD является полезным инструментом для анализа и проектирования системы, позволяя визуализировать и изучить ее структуру и потоки данных. Он может быть использован для выявления проблемных мест, оптимизации процессов и обнаружения возможностей для автоматизации и улучшений в системе.

## 2.2 Проектирование средствами UML

Диаграмма вариантов использования – это графическое представление функциональных возможностей системы, которые могут быть использованы ее пользователями. Она состоит из двух основных элементов: актеров (Actor) и вариантов использования (Use Case). Актеры – это пользователи системы, которые могут взаимодействовать с ней, а варианты использования описывают, какие функциональности системы доступны для пользователей.

Построим диаграмму вариантов использования для рассмотренной в семестровом проекте информационной системы учета арендной платы. Система имеет два класса пользователей – администратор и менеджер. С проектируемой системой взаимодействуют актер «администратор» и актер «менеджер».

Диаграмма вариантов использования представлена в соответствии с рисунком 2.3.



Рисунок 2.3 Диаграмма вариантов использования

Базовыми вариантами использования для актера «Администратор» будут «Авторизация», «Взаимодействие со справочником «Объекты», «Взаимодействие со справочником «Арендаторы», «Взаимодействие со справочником «Договоры», «Взаимодействие со справочником «Пользователи» и «Просмотр статистики». «Взаимодействие со справочником «Объекты» включает в себя «Добавление объектов», «Удаление объектов» и «Редактирование». «Взаимодействие со справочником «Арендаторы» включает в себя «Добавление арендаторов», «Удаление арендаторов» и «Редактирование арендаторов». «Взаимодействие со справочником «Договоры» включает в себя «Добавление договоров», «Удаление договоров» и «Редактирование договоров». «Взаимодействие со справочником «Пользователи» включает в себя «Добавление пользователей», «Удаление пользователей» и «Редактирование пользователей».

Базовыми вариантами использования для актера «Менеджер» будут «Авторизация», «Взаимодействие со справочником «Объекты», «Взаимодействие со справочником «Арендаторы», «Взаимодействие со справочником «Договоры» и «Просмотр статистики». «Взаимодействие со справочником «Объекты» включает в себя «Добавление объектов», «Удаление объектов» и «Редактирование». «Взаимодействие со справочником «Арендаторы» включает в себя «Добавление арендаторов», «Удаление арендаторов» и «Редактирование арендаторов». «Взаимодействие со справочником «Договоры» включает в себя «Добавление договоров», «Удаление договоров» и «Редактирование договоров».

Диаграмма классов отображает классы, их атрибуты (переменные) и методы (функции), а также отношения между классами. На диаграмме классов каждый класс представлен прямоугольником с названием класса внутри. Атрибуты и методы класса могут быть представлены внутри этого прямоугольника или над ним. Связи между классами отображаются в виде линий с различными аннотациями [7, 8].

Диаграмма классов позволяет описать структуру программной системы и ее компонентов, а также показать связи между ними.

Диаграмма классов представлена в соответствии с рисунком 2.4.



Рисунок 2.4 Диаграмма классов

Построение диаграммы классов для рассмотренной в семестровом проекте системы для учета арендной платы. Статические модели обеспечивают представление структуры системы в терминах базовых строительных блоков и отношений между ними. Диаграмма классов служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма классов может отражать, в частности, различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывает их внутреннюю структуру и типы отношений. На данной диаграмме не указывается информация о временных аспектах функционирования системы. Статическая модель системы должна содержать классы-сущности (стереотип «entity») для работы с таблицами базы данных.

Диаграмма состояний процесса «Оформление нового договора» в соответствии с рисунками 2.5-2.6.

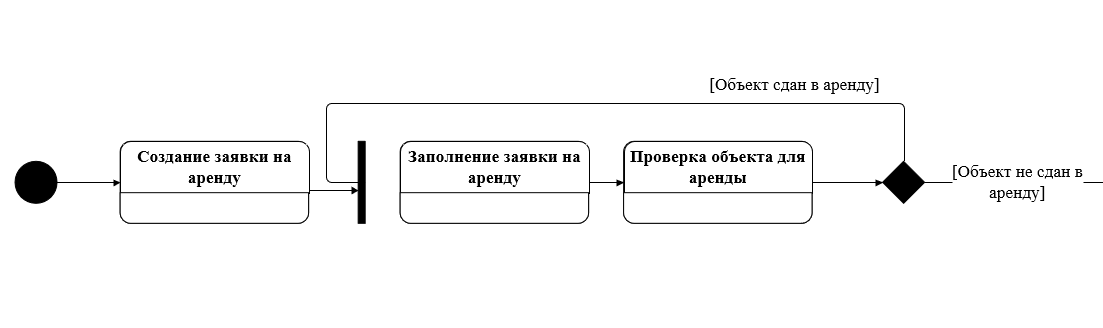


Рисунок 2.5 Диаграмма состояний (часть 1)

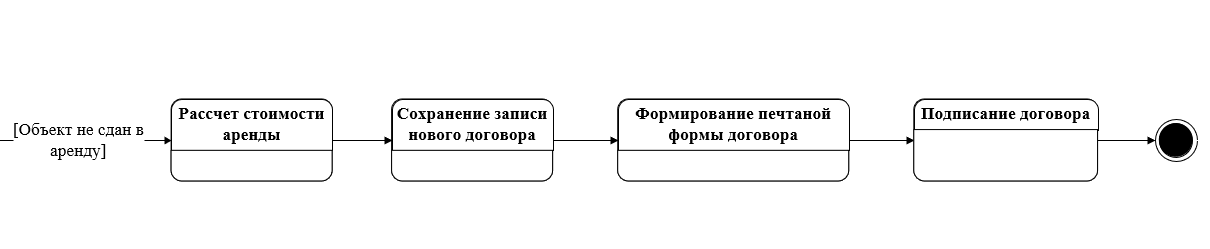


Рисунок 2.6 Диаграмма состояний (часть 2)

Отношения «Ассоциация»– представляет собой отношения между экземплярами классов.

Отношения «Агрегация»– это ассоциация типа «целое-часть».

Отношения «Композиция» – это такая агрегация, где объекты-части не могут существовать сами по себе и уничтожаются при уничтожении объекта агрегирующего класса.

Отношения «Наследование»– это отношение типа «общее-частное». Позволяет определить такое отношение между классами, когда один класс обладает поведением и структурой ряда других классов. При создании производного класса на основе базового (одного или нескольких) возникает иерархия наследования.

Описать возможные состояния и переходы, которые в совокупности характеризуют поведение элемента модели в течение его жизненного цикла можно показать на диаграмме состояний. Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий [9]. Диаграмма строится для отдельного класса, варианта использования, отдельной операции класса или целой подсистемы. В данной диаграмме начальное состояние представляет собой частный случай состояния, которое не содержит никаких внутренних действий.

На диаграмме можно увидеть состоянии процесса и последовательные подсостояние. Последовательные подсостояния – это вложенные состояния состояния-композита, в рамках которого в каждый момент времени объект может находиться в одном и только одном подсостоянии. Поведение объекта в этом случае представляет собой последовательную смену подсостояний, от начального до конечного.

Диаграмма состояний для процесса оформления нового договора позволяет визуализировать и понимать, как происходит взаимодействие пользователей с системой учета арендной платы и какие состояния процесса возникают.

При моделировании поведения проектируемой или анализируемой системы возникает необходимость не только представить процесс изменения ее состояний, но и детализировать особенности алгоритмической и логической реализации выполняемых системой операций. Для моделирования процесса выполнения операций в языке UML используются так называемые диаграммы деятельности. При построении диаграммы деятельности используются только не триггерные переходы, т.е. такие, которые срабатывают сразу после завершения деятельности или выполнения соответствующего действия. Этот переход переводит деятельность в последующее состояние сразу, как только закончится действие в предыдущем состоянии. На диаграмме такой переход изображается сплошной линией со стрелкой. Графически ветвление на диаграмме деятельности обозначается небольшим ромбом, внутри которого нет никакого текста.

Первым алгоритмом является авторизация, а затем происходит открытие справочника арендаторов. Далее пользователь, если не находит нужного арендатора, то добавлять новую запись об арендаторе. Для этого открывает форму, заполняет необходимые поля, после чего они отправляются в программу, оттуда идет запрос к БД на сохранение данных. После того, как данные добавлены, пользователь открывает справочник, содержащий договора, открывает и заполняет форму для создания нового договора, далее программа отправляет запрос к БД на создание новой записи, после создания записи, программа формирует печатную форму договора и отправляет результат пользователю. Результатом будет сохраненная запись нового договора и печатная форма, после чего процесс завершается.

Диаграмма деятельности представлена в соответствии с рисунком 2.7.



Рисунок 2.7 Диаграмма деятельности

Диаграмма последовательности – это один из видов диаграмм, которые используются для описания взаимодействия между компонентами или объектами в программной системе. Она отображает последовательность сообщений, передаваемых между компонентами или объектами во времени.

Диаграмма последовательности часто используется для описания взаимодействия между компонентами или объектами в рамках одного процесса, таких как взаимодействие между компонентами веб-приложения или между классами в программе [10, 11]. Она позволяет легко визуализировать порядок передачи сообщений между компонентами или объектами и помогает разработчикам понять логику взаимодействия в системе.

Диаграмма последовательности для данного проекта представлена в соответствии с рисунком 2.8.



Рисунок 2.8 Диаграмма последовательности

Диаграмма компонентов – это визуальное представление архитектуры разрабатываемой системы, которая показывает компоненты и зависимости между ними. Компоненты могут быть программными, аппаратными или любыми другими элементами системы [12]. На диаграмме компонентов компоненты обычно представляются в виде прямоугольников, а зависимости между ними – стрелками.

Пунктирные стрелки, соединяющие модули, показывают отношения взаимозависимости, аналогичные тем, которые имеют место при компиляции исходных текстов программ.

Диаграмма компонентов ИС для учета арендной платы представлена в соответствии с рисунком 2.9.



Рисунок 2.9 Диаграмма компонентов

Диаграмма компонентов позволяет лучше понимать архитектуру приложения, определять зависимости между компонентами и легче проводить тестирование и сопровождение приложения.

## 2.3 Проектирование баз данных

ER-диаграмма, также известная как схема «сущность-связь» или ERD, представляет собой тип блок-схемы, который показывает, как различные объекты или концепции связаны внутри системы [11, 13]. Для отображения сущностей, их атрибутов и связей ERD-модели используют стандартный набор символов, таких как прямоугольники, ромбы, овалы и соединительные линии. Аналогично грамматическим структурам, сущности выполняют роль существительных, а связи – глаголов.

Прямоугольники используются для обозначения объектов или концепций, которые содержат важную информацию и называются «сущностями».

Ромбы используются для обозначения связей между двумя сущностями, которые выражаются глаголами.

Атрибуты ERD характеризуют сущности и содержат информацию о них. Они обозначаются овалами и помогают пользователям лучше понимать устройство базы данных.

В ER-диаграмме АИС для учета арендной платы «Альфа» определены такие сущности: «Объект», «Арендатор (физ. лица)», «Арендатор (юр. лица)», «Договор (физ. лица)», «Договор (юр. лица)» и «Пользователь».

Связи: «заключает» и «включает».

ER-диаграмма представлена в соответствии с рисунком 2.10.



Рисунок 2.10 ER-диаграмма

Даталогическая модель базы данных определяет, как сущности и связи, описанные в инфологической модели, будут реализованы с учетом особенностей используемой СУБД и ее ограничений. В ней описываются таблицы, поля и связи между ними, а также ограничения на данные.

Каждая таблица представляет отдельную сущность и имеет уникальное имя и набор полей, описывающих свойства этой сущности. Связи между таблицами определяют отношения между сущностями.

Даталогическая модель также учитывает ограничения на таблицы и связи, такие как уникальность значений и ограничения на ввод и удаление данных. Типы данных для каждого поля и связи определяются в модели, а также индексы, которые используются для ускорения доступа к данным.

Для проекта «Альфа», даталогическая модель будет содержать таблицы, такие как «Объекты», «Арендаторы (физ. лица)», «Арендаторы (юр. лица)», «Договоры (физ. лица)», «Договоры (юр. лица)» и «Пользователи». Каждая таблица будет содержать свои поля, ограничения и связи с другими таблицами.

Даталогическая модель представлена в соответствии с рисунком 2.11.



Рисунок 2.11 Даталогическая модель

Даталогическая модель предоставляет возможность оптимизировать работу с данными и обеспечить целостность и надежность базы данных.

Преимущество даталогической модели заключается в более точном определении структуры базы данных и предотвращении возможных проблем согласованности данных. Она также упрощает процесс создания физической модели базы данных и повышает ее эффективность в дальнейшей работе.

## 2.4 Проектирование пользовательских интерфейсов

Проектирование пользовательских интерфейсов является процессом создания визуальной и функциональной компоненты, которые пользователи будут использовать для взаимодействия с приложением [13, 14].

При открытии информационной системы в первую очередь происходит открытие окна для входа в программу. Пользователю необходимо ввести свои данные (логин и пароль), чтобы продолжить работу с программой.

Окно авторизации в соответствии с рисунком 2.12.

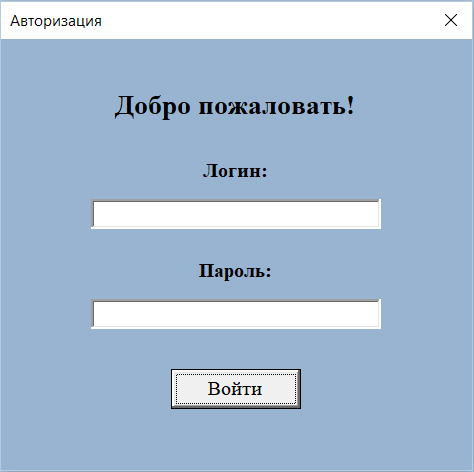


Рисунок 2.12 Окно авторизации

После успешного входа в программу открывается главная страница с навигационным меню. Навигационное меню содержит 3 главных раздела: «Справочники», «Отчеты» и «Параметры». Также на главной странице имеется кнопка для выхода, которая перенаправляет пользователя на форму авторизации. Главная страница представлена в соответствии с рисунком 2.13.

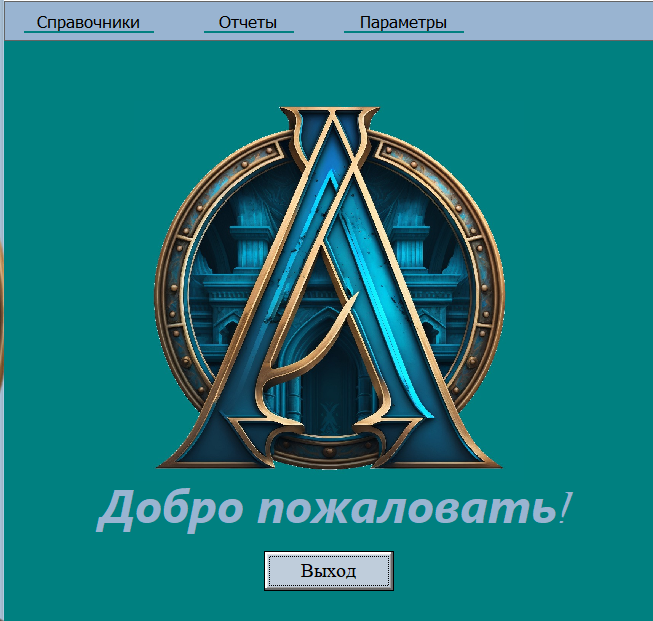


Рисунок 2.13 Главная страница системы

Каждый раздел навигационного меню содержит выпадающее меню. Раздел «Справочники» содержит следующие пункты подменю:

* договоры – содержит информацию об договорах физических и юридических лиц;
* объекты – содержит информацию о помещениях организации, которые сдаются в аренду;
* арендаторы – содержит информацию об арендаторах компании, подразделяется на физических и юридических лиц;
* пользователи – содержит информацию о пользователях программы, данный справочник доступен только для администратора.

При нажатии на «Объекты» открывается форма соответствующего справочника. Выводится полная информация обо всех объектах данной организации, которые сдаются в аренду [15].

Форма справочника «Объекты» представлена в соответствии с рисунком 2.14.

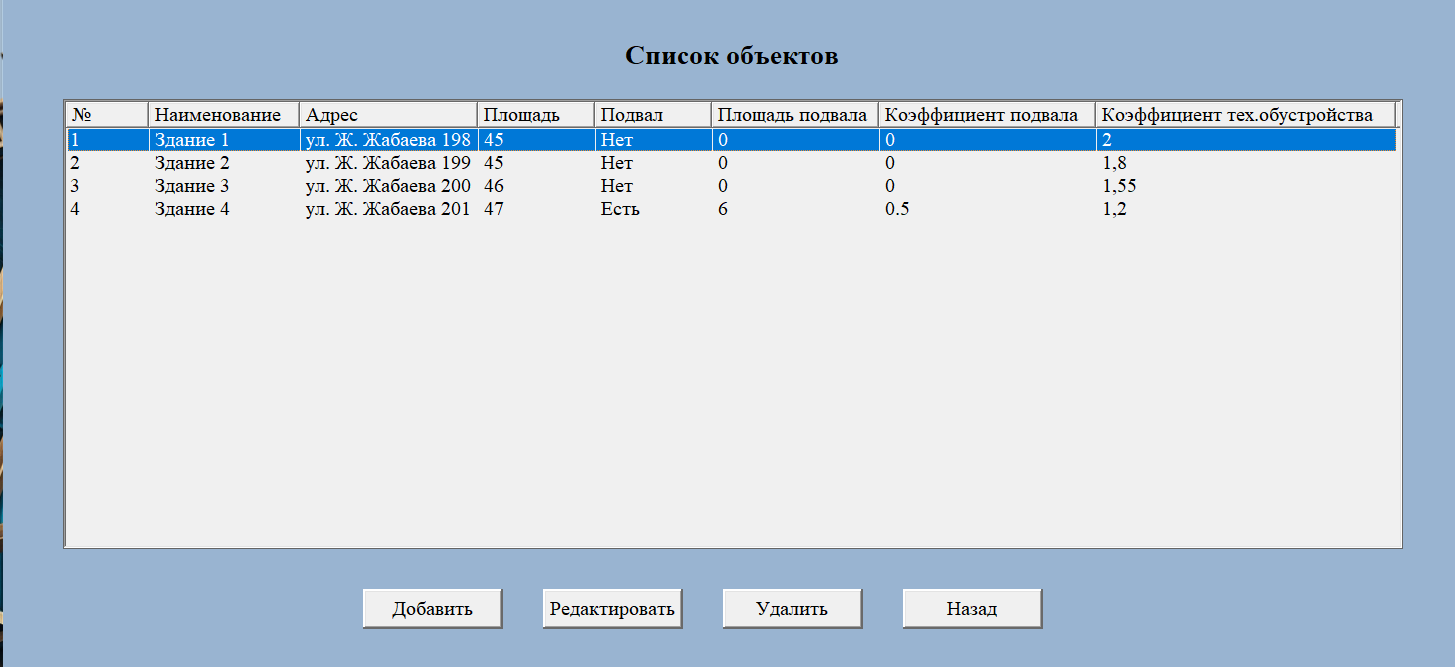


Рисунок 2.14 Справочник «Объекты»

Данная форма имеет кнопки «Добавить», «Редактировать», «Удалить» и «Назад». Кнопка «Редактировать» позволяет пользователю внести изменения в выбранный объект, все изменения автоматически вносятся в базу данных. Кнопка «Удалить» позволяет удалить запись о выбранном объекте из БД. При нажатии на кнопку «Добавить» открывается форма для добавления информации о новом помещение.

Данная форма имеет необходимые поля для заполнения, а также кнопку «Подвал», при активизации данного выбора открываются два дополнительных поля для заполнения: «Площадь подвала» и «Коэффициент подвала», в случае, если галочка не будет установлена, то поля видны не будут. Также на форме есть две кнопки: «Сохранить» и «Очистить» [16]. Которые соответственно сохраняют новые данные в БД, а вторая очищает поля в случае, если они были заполнены неверно.

Форма для добавления нового объекта представлена в соответствии с рисунком 2.15.

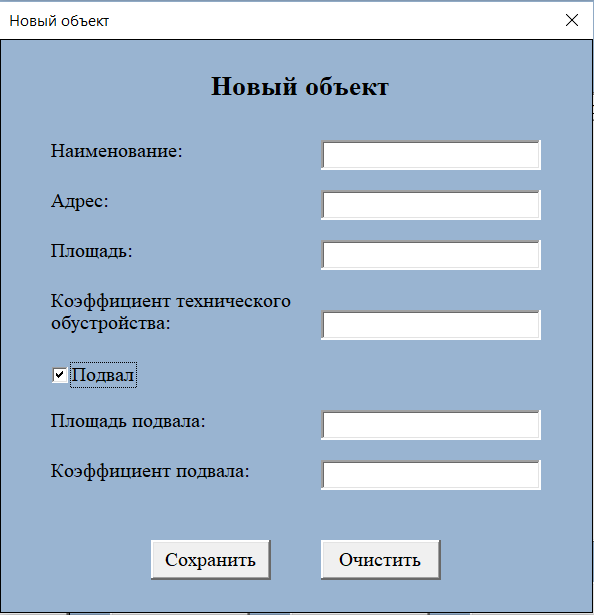


Рисунок 2.15 Форма добавления нового объекта

При нажатии на «Арендаторы» пользователю выпадет меню для выбора типа арендатора. В зависимости от выбранного типа будет показана соответствующая страница.

Справочник «Арендаторы (физ.лица)» представлен в соответствии с рисунком 2.16.

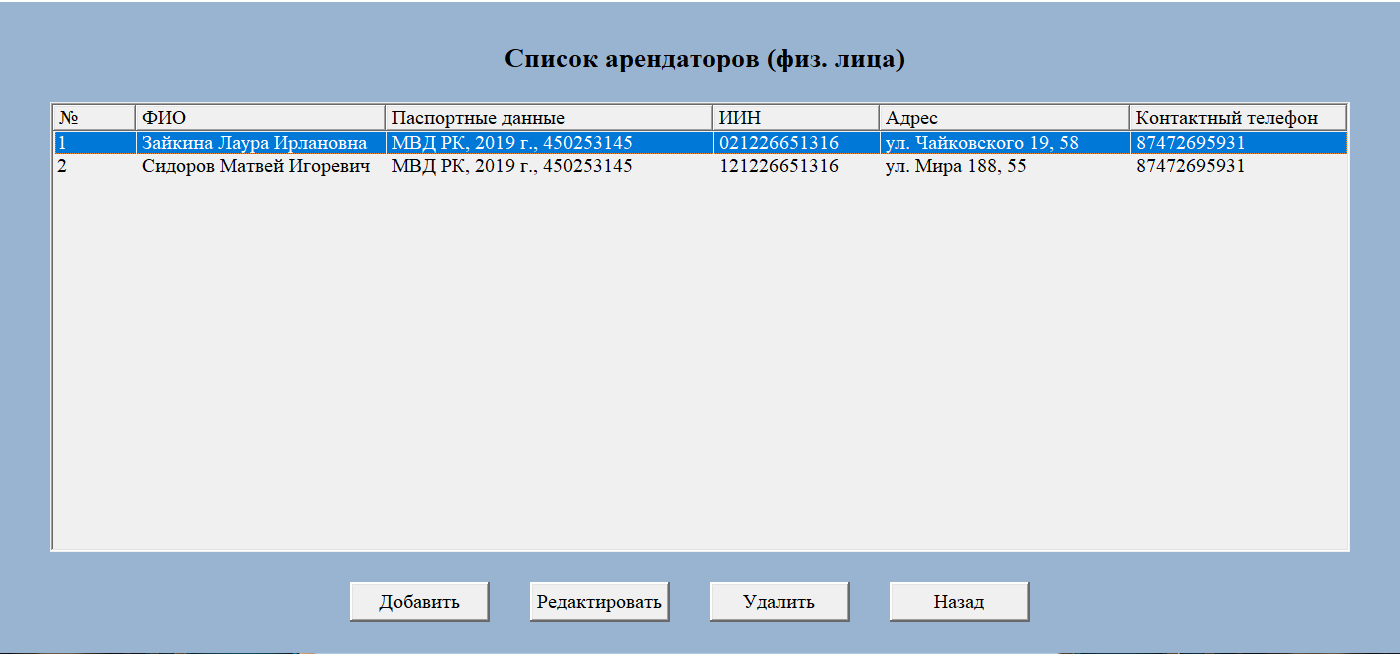


Рисунок 2.16 Справочник «Арендаторы (физ.лица)»

Данная форма имеет кнопки «Добавить», «Редактировать», «Удалить» и «Назад». Кнопка «Редактировать» позволяет пользователю внести изменения в выбранную запись об арендаторе, все изменения автоматически вносятся в базу данных. Кнопка «Удалить» позволяет удалить запись о выбранном арендаторе из БД. При нажатии на кнопку «Добавить» открывается форма для добавления информации о новом арендаторе. Форма для добавления нового арендатора представлена в соответствии с рисунком 2.17.



Рисунок 2.17 Форма добавления нового арендатора (физ.лицо)

Справочник «Арендаторы (юр.лица)» представлен в соответствии с рисунком 2.18.

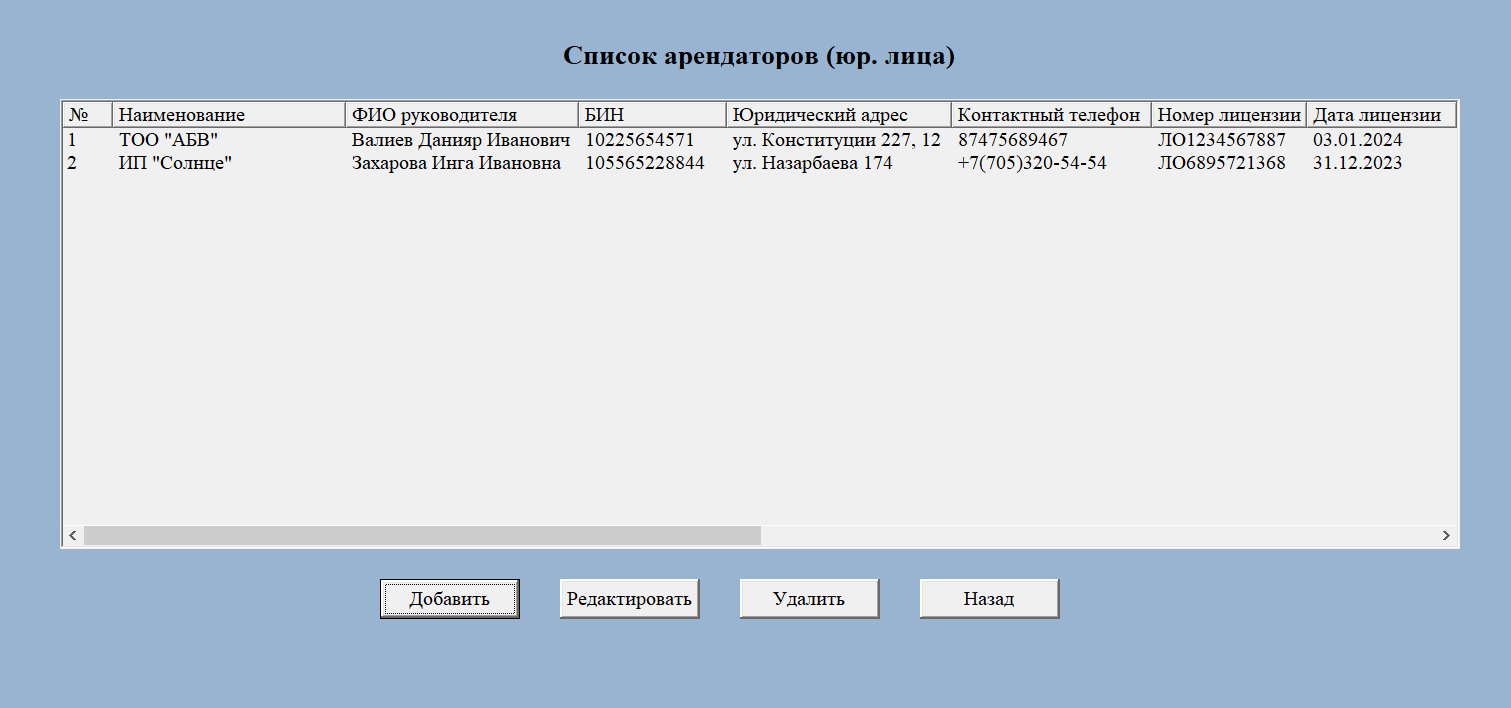


Рисунок 2.18 Справочник «Арендаторы (юр.лица)»

Данная форма имеет такие же кнопки, как и справочник «Арендаторы (физ.лица)». Но форма добавления нового арендатора отличается, так как юридическое лицо имеет другие пункты для заполнения. Форма для добавления нового арендатора представлена в соответствии с рисунком 2.19.

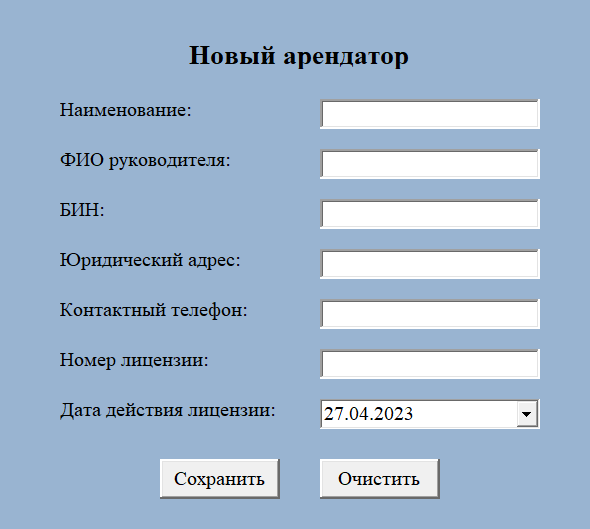


Рисунок 2.19 Форма добавления нового арендатора (юр.лицо)

Следующий раздел «Договоры» также подразделяется на 2 типа в зависимости от выбранного варианта. Имеются договора для физических лиц, а есть договора для юридических лиц.

Справочник «Договоры (физ.лица)» представлен в соответствии с рисунком 2.20.



Рисунок 2.20 Справочник «Договоры (физ.лица)»

Форма справочника содержит точно такой же функционал, как и другие справочники. Форма добавления нового договора представлена в соответствии с рисунком 2.21.

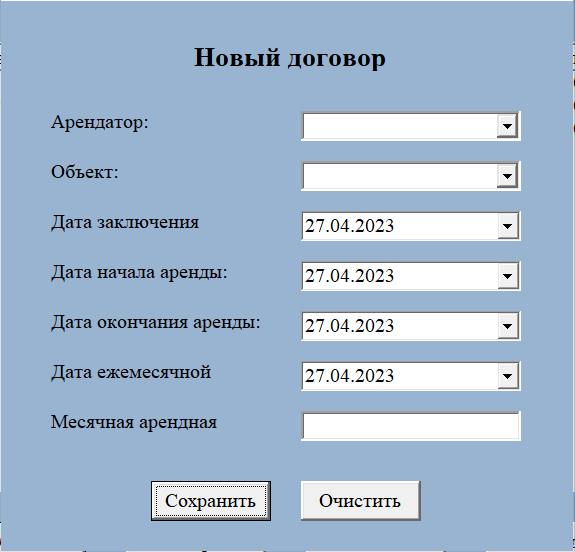


Рисунок 2.21 Форма добавления нового договора (физ.лицо)

Данная форма содержит поле «Арендатор», в котором содержаться все арендаторы по ИИНу из таблицы арендаторов, то есть можно выбрать из выпадающего списка, изучая весь список, либо ввести данные самим и выбрать, если такой арендатор существует. В случае отсутствия арендатора, необходимо сначала создать запись о нем. Затем идет поле «Объект», где также необходимо выбрать из имеющихся объектов организации. При этом при сохранении договора будет производится проверка, что данный объект в выбранный период свободен для сдачи в аренду. Если объект не свободен, договор не сохранится и выведется сообщение об ошибке.

Месячная арендная плата заполняется автоматически на основе выбранного объекта и вычисляется по формуле: МАП = (базовая ставка/12 \* площадь помещения + базовая ставка/12 \* площадь подвала \* коэффициент подвала) \* КТ. Где базовая ставка является числом, установленным законодательством РК [17].

При сохранении нового договора формируется заполненный договор по шаблону в Word.

Справочник «Договоры (юр.лица)» представлен в соответствии с рисунком 2.22.

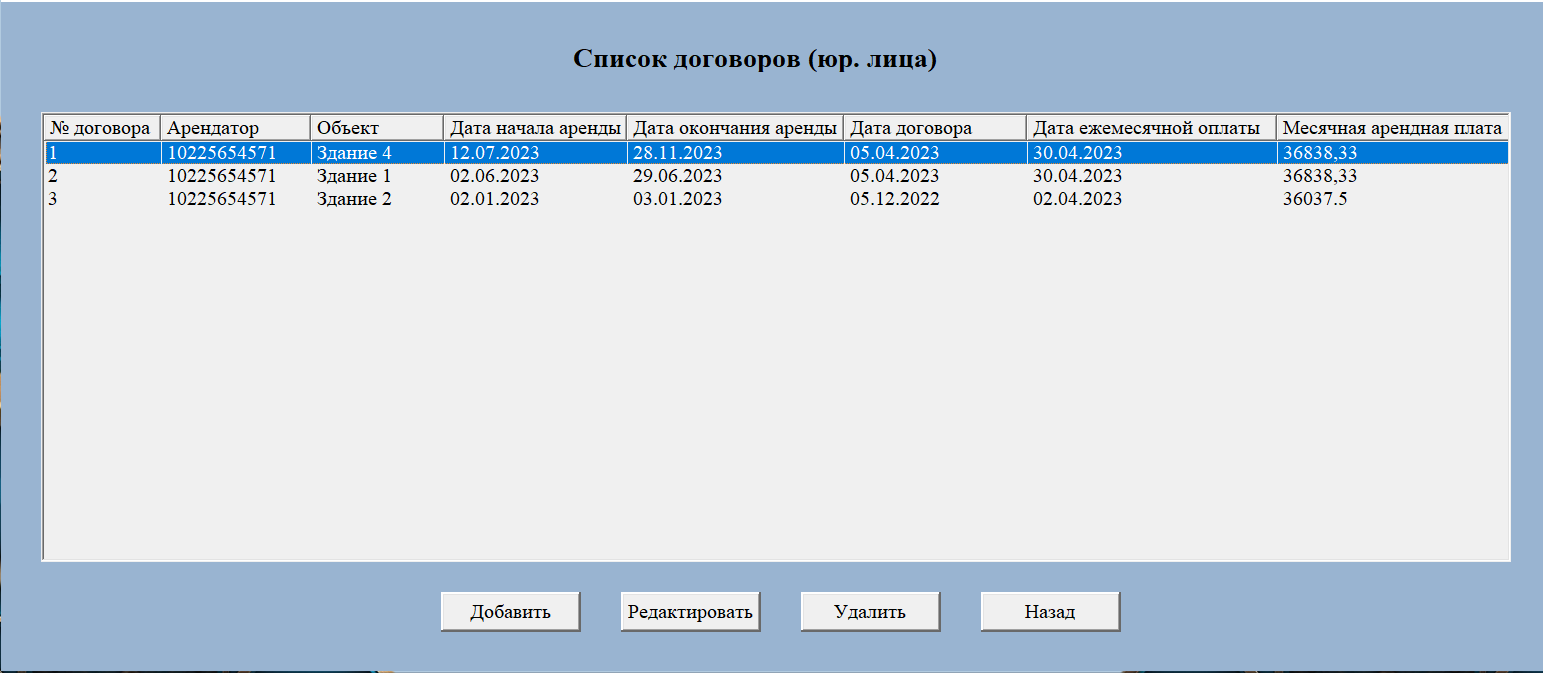


Рисунок 2.22 Справочник «Договоры (юр.лица)»

Форма справочника содержит точно такой же функционал, как и другие справочники. Форма добавления нового договора представлена в соответствии с рисунком 2.23.

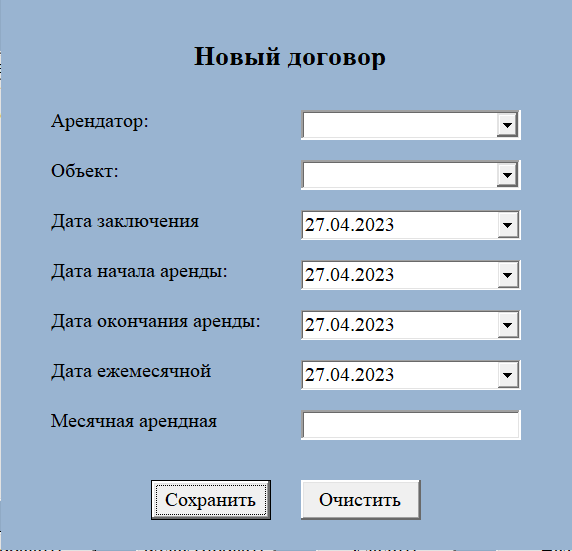


Рисунок 2.23 Форма добавления нового договора (юр.лицо)

Данная форма содержит поле «Арендатор», в котором содержаться все арендаторы по БИНу из таблицы арендаторов, то есть можно выбрать из выпадающего списка, изучая весь список, либо ввести данные самим и выбрать, если такой арендатор существует. В случае отсутствия арендатора, необходимо сначала создать запись о нем. Все остальные поля идентичны полям из формы создания договоров для физических лиц. При сохранении нового договора формируется заполненный договор по шаблону в Word.

Для администратора доступен справочник «Пользователи». Подразумевается, что администратор имеет возможность добавлять новых сотрудников компании. Форма справочника «Пользователи» представлен в соответствии с рисунком 2.24.

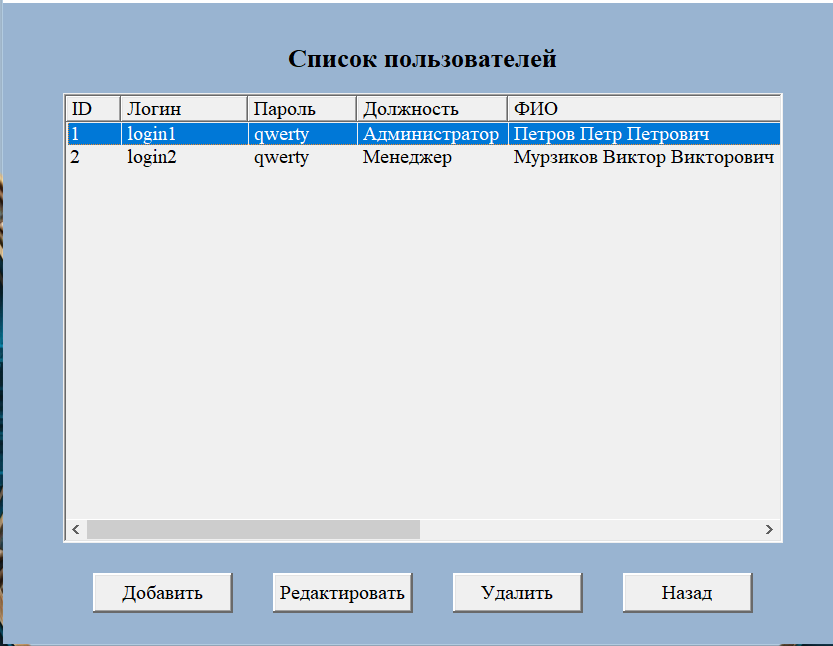


Рисунок 2.24 Справочник «Пользователи»

Форма добавления нового пользователя информационной системы представлена в соответствии с рисунком 2.25.

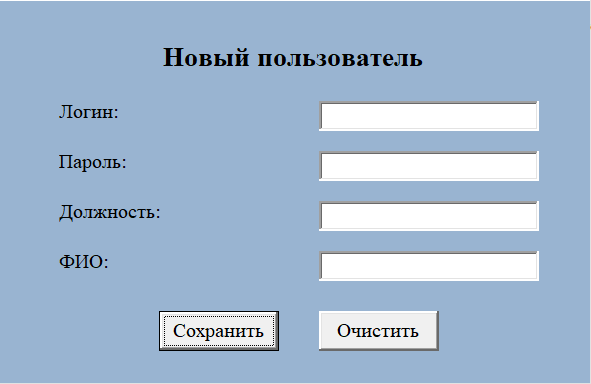


Рисунок 2.25 Форма добавления нового пользователя

Следующим разделом навигационного меню является раздел «Отчеты». Данный раздел необходим для аналитики работы компании. Раздел содержит следующие подпункты:

* отчет о прибыли – показывает сумму прибыли на выбранный месяц;
* отчет об арендаторах – показывает действующих арендаторов на выбранную дату;
* отчет об объектах – показывает информацию об объектах, которые не сланы в аренду на выбранную дату.

«Отчет о прибыли» является финансовым документом, который отображает сумму прибыли, полученной компанией за определенный период времени. В данном случае, отчет отображает сумму прибыли на выбранный месяц. Данный отчет может использоваться компанией для анализа ее финансового состояния и принятия решений о дальнейшей стратегии развития. Причем формироваться отчет может только на текущий месяц, либо на прошлые месяцы. Отчет о прибыли представлен в соответствии с рисунком 2.26.

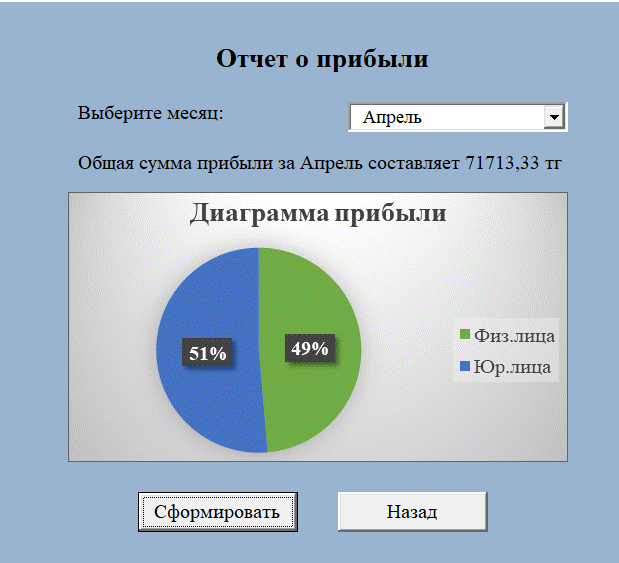


Рисунок 2.26 Отчет о прибыли

«Отчет об арендаторах» является документом, который отображает, список действующих арендаторов на выбранную дату. В данном случае, отчет отображает также информацию об арендаторе, его типе, и название объекта, который он арендует. Также имеется гистограмма, которая отражаете, какое количество объектов арендует определенный арендатор. Отчет об арендаторах представлен в соответствии с рисунком 2.27.

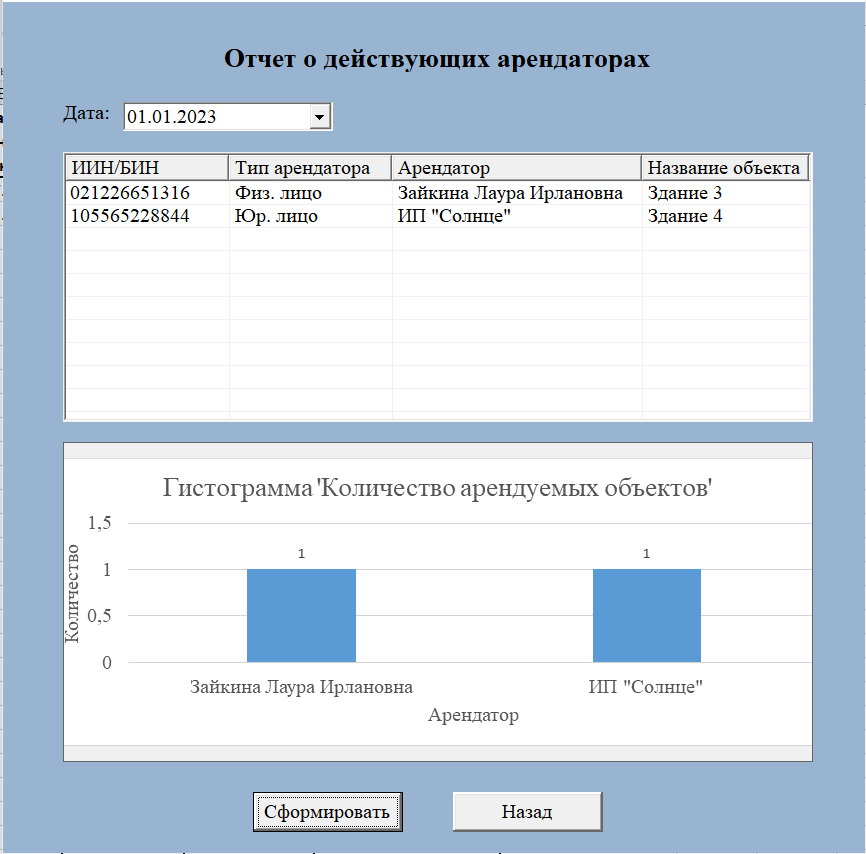


Рисунок 2.27 Отчет об арендаторах

«Отчет об объектах» является документом, который отображает, список помещений, которые не сданы в аренду на выбранную дату. Отчет о неарендованных объектах представлен в соответствии с рисунком 2.28.



Рисунок 2.28 Отчет о неарендованных объектах

Последний раздел навигационного меню «Параметры» имеет только 1 подпункт «Разработчики…», при нажатии на который открывается сообщение с информационное сообщение. Информация о разработчиках представлена в соответствии с рисунком 2.29.

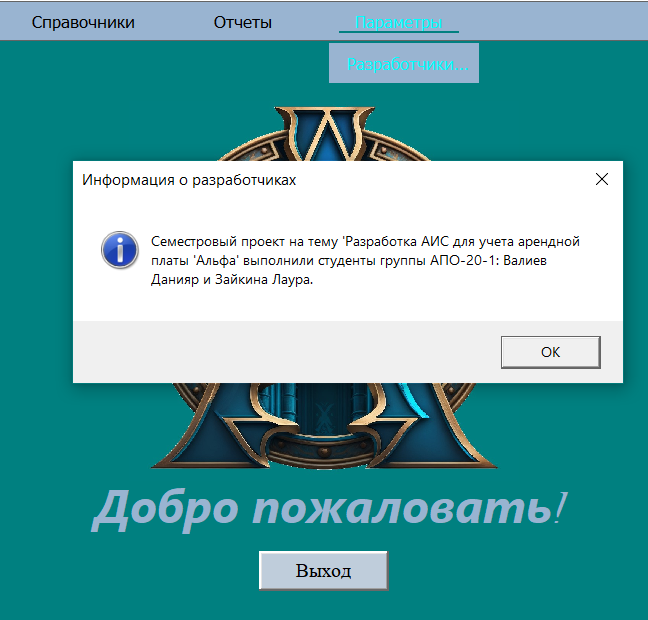


Рисунок 2.29 Информация о разработчиках

Проектирование пользовательского интерфейса – это сложный процесс, требующий знаний и опыта в области дизайна и технологий. Однако, при правильном подходе к этому процессу, результат может быть положительным, обеспечивая пользователям легкий и приятный доступ к функционалу приложения.

# ЗАКЛЮЧЕНИЕ

В результате семестрового проекта была разработана информационная система для учета арендной платы организации. Данный ресурс предназначен для компаний, которые хотят автоматизировать бизнес-процессы и упростить ведение учета. Сотрудники смогут с легкостью отслеживать данные по договорам, формировать печатные формы на основе введенных данных, экономя свое время и время клиента, а также теперь пользователям доступны отчеты для аналитики работы предприятия.

В результате реализации семестрового проекта были выполнены поставленные цели и задачи. Ресурс имеет четкую логическую структуру и удобную навигацию. Вся информация, находящаяся в информационной системе, размещается в базе данных на листах Excel. В процессе работы над проектом были использованы современные технологии и методы, чтобы создать удобный и простой в использовании интерфейс для пользователей.

Для разработки использовались такие технологии как:

* VBA;
* Microsoft Excel;
* Microsoft Word.

На данный момент инормационная система предоставляет следующий функционал:

* просмотр справочника объектов, с возможностью редактирования;
* просмотр справочника арендаторов, с возможностью редактирования;
* просмотр справочника договоров, с возможностью редактирования;
* создание новых договоров, с автоматической печатью печатных форм данного договора в соответствие со стандартным шаблоном;
* формирование отчетов для аналитики работы предприятия.

Семестровый проект выполнен в светлой и спокойной цветовой гамме с индивидуальным логотипом. В информационной системе реализована возможность управления пользователями. Администратор имеет возможность добавлять новых сотрудников, редактировать информацию о действующих или удалять недействительных работников для защиты данных организации.

В завершение можно отметить, что разработка информационной системы для учета арендной платы за нежилые помещения обеспечивает доступность и удобство для пользователей. Данный проект может стать полезным и востребованным инструментом для многих компаний, которые хотят автоматизировать и упростить свой бизнес, при этом не затрачивая огромные ресурсы на разработку.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Ушакова, Е.В. Методические указания по выполнению дипломных проектов/работ для бакалавриата направления информационно-коммуникационные технологии [Текст] / Е.В. Ушакова, Л.В. Долматова. –Петропавловск: СКУ им. М.Козыбаева, 2020. – 99 с.;
2. Учебник VBA [Электронный ресурс] – Режим доступа: URL: http://is.ku.edu.kz/methodpages/green/ISRP/frizen.zip (02.05.2023);
3. Плакунов М. К., Планирование на малых и средних предприятиях средствами EXCEL [Текст] / Плакунов М. К. – СПб.: Питер, 2018. – 184 с.;
4. Гарбер Г. Основы програмирования на Visual Basic и VBA в Excel 2007 [Текст] / Г.Гербер. – М.: АСТ, 2017. – 319 с.;
5. Уокенбах Д. Excel 2016: профессиональное программирование на VBA [Текст] / Д. Уокенбах. – М: ДИАЛЕКТИКА, 2016. – 944 с.;
6. Журавлев А.В. Excel 2016. Программирование на VBA [Текст] / Журавлев А.В. – М.: Лань, 2017. – 320 с.;
7. Гарнаев А.Ю., Самоучитель VBA [Текст] / А.Ю. Гарнаев. – СПб: БХВ-Петербург, 2014. – 560 с.;
8. Покровский А.В., Шевелева Е.А. Программирование в VBA для Excel 2013 [Текст]: учебное пособие / А.В. Покровский, Е.А. Шевелева. – СПб.: БХВ-Петербург, 2014. – 336 с.;
9. Басков Н.Н., Лебединская А.Р. Visual Basic for Applications Учебный практикум [Текст] / Н.Н. Басков, А.Р. Лебединская. – Ростов на Дону: РИФ РГТЭУ, 2018. – 86 с.;
10. Культин Н.Б., Цой Л.Б. VBA для студентов и школьников [Текст] / Н.Б. Культин, Л.Б. Цой. – М.: Юрайт , 2020. – 513 с.;
11. МакГрат М. Excel VBA. Стань продвинутым пользователем за неделю [Текст] / М. МакГрат. – М.: Бомбора, 2019. – 237с.;
12. Комолова Н., Клименко А. Программирование на VBA в Excel 2019 [Текст] / Н. Комолова, А. Клименко. – М.: БХВ-Петербург, 2020. – 497 с.;
13. Ракитов А.А. Программирование на VBA в приложениях Microsoft Office 2013 [Текст]: учебное пособие / А.А. Ракитов. – М.: Юрайт, 2015. – 256 с.;
14. Туркин О.В. VBA. Практическое программирование [Текст] / О.В. Туркин. – М.: СОЛОН-Пресс, 2016. – 127 с.;
15. Швыдков С.В. Учебник по функциям Excel и программированию в среде VBA [Текст]: учебное пособие / С.В. Швыдков. – М.: Изд-во МГУ им. М.В. Ломоносова, 2019. – 105 с.;
16. Михеев Р. VBA и программирование в MS Office для пользователей [Текст] / Р. Михеев. – М.: БХВ-Петербург, 2016. – 373 с.;
17. Арсентьев А.В. Microsoft Excel VBA [Текст] / А.В. Арсентьев. – М.: ДМК Пресс, 2017. – 256 с.