

**Министерство науки и высшего Российской Федерации**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  
**(ФГБОУ ВО «АмГУ»)**

Факультет математики и информатики  
Кафедра информационных и управляющих систем  
Направление подготовки 09.03.01 – Информатика и вычислительная техника  
Направленность (профиль) образовательной программы Автоматизированные системы обработки информации и управления

**ДОПУСТИТЬ К ЗАЩИТЕ**  
Зав. кафедрой

\_\_\_\_\_ А.В. Бушманов  
«\_\_\_» \_\_\_\_\_ 2023 г.

**БАКАЛАВРСКАЯ РАБОТА**

на тему: Разработка автоматизированной системы управления общежитием для  
ФГБОУ ВО «АмГУ»

Исполнитель  
студент группы 953об

\_\_\_\_\_

(подпись, дата)

М.Д. Яковлев

Руководитель  
доцент, канд. техн.  
наук.

\_\_\_\_\_

(подпись, дата)

А.В. Бушманов

Консультант:  
по части безопасности  
и экологичности  
доцент, канд. техн. наук

\_\_\_\_\_

(подпись, дата)

А.Б. Булгаков

Нормоконтроль  
инженер кафедры

\_\_\_\_\_

(подпись, дата)

В.Н. Адаменко

**Министерство науки и высшего образования Российской Федерации**  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
**АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  
**(ФГБОУ ВО «АмГУ»)**

Факультет Математики и информатики

Кафедра Информационных и управляющих систем

УТВЕРЖДАЮ

Зав. Кафедрой

\_\_\_\_\_ А.В. Бушманов

« \_\_\_\_ » \_\_\_\_\_ 2023г

**ЗАДАНИЕ**

К выпускной квалификационной работе студента: Яковлев М.Д.

1. Тема выпускной квалификационной работы: Разработка автоматизированной системы управления общежитием для ФГБОУ ВО «АмГУ»

(утверждена приказом 24.04.2023 №974-уч)

2. Срок сдачи студентом законченной работы (проекта): 20.06.2023

3. Содержание выпускной квалификационной работы: анализ предметной области студенческого городка, проектирование архитектуры АСУ, разработка базы данных, программная реализация АСУ, обеспечение информационной безопасности системы, обоснование мер по обеспечению безопасности и экологичности использования системы.

4. Перечень материалов приложения: скриншоты пользовательского интерфейса программы, техническое задание к разработке программы

5. Дата выдачи задания: 30.01.2023

Руководитель выпускной квалификационной работы: \_\_\_\_\_

Бушманов А.В. заведующий кафедры ИиУС, канд. техн. наук, доцент

(фамилия, имя, отчество, должность, уч. степень, уч. звание)

Задание принял к исполнению: 30.01.2022 \_\_\_\_\_

Яковлев М.Д.

## РЕФЕРАТ

Бакалаврская работа содержит 103 с., 36 таблиц, 49 рисунков, 23 библиографических источника, 5 приложений

АНАЛИЗ ДЕЯТЕЛЬНОСТИ ПРЕДПРИЯТИЯ, АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ, БАЗА ДАННЫХ, ИНФОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ, ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ, ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ, POSTGRE SQL, MICROSOFT VISUAL STUDIO 2022, WPF, EXCEL, ИНТЕРФЕЙС, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ, ТРЕБОВАНИЯ, РЕАЛИЗАЦИЯ, ОБЩЕЖИТИЕ.

Целью бакалаврской работы является разработка автоматизированной системы управления общежитием для ФГБОУ ВО «АмГУ». В процессе работы были изучены процессы и деятельность общежития, а затем разработана система для упрощения этих процессов и повышения эффективности управления.

Для достижения поставленной цели были выполнены следующие задачи:

- изучение литературы по теме управления общежитием и информационных систем;
- анализ бизнес-процессов общежитий ФГБОУ ВО «АмГУ»;
- исследование информационных потоков, происходящих в общежитии;
- разработка концептуальной и логической модели данных для системы управления общежитием;
- разработка и реализация программного обеспечения для системы;

В процессе разработки работы были использованы материалы периодической печати, учебные материалы, а также электронные ресурсы. Использовались программные средства: Microsoft Office Word, Microsoft Visual Studio 2022, PostgreSQL

## СОДЕРЖАНИЕ

Сокращения	6
Нормативные ссылки	7
Введение	9
1 Анализ деятельности студенческого городка ФГБОУ ВО «АмГУ»	10
1.1 Описание студенческого городка ФГБОУ ВО «АмГУ»	10
1.2 Функции общежитий	11
1.3 Организационная структура студенческого городка	12
1.4 Описание потоков данных и бизнес-процессов в общежитиях	14
1.5 Обзор и анализ существующих проектных решений, выявление до- стоинств, недостатков	18
1.6 Обоснование необходимости разработки АСУ	22
2 Проектирование архитектуры построения и выбора платформы реали- зации АСУ	24
2.1 Выбор архитектуры построения АСУ	24
2.2 Технические характеристики ПК-клиента АСУ	26
2.3 Обеспечивающие и функциональные подсистемы АСУ	28
3 Проектирование автоматизированной системы управления общежи- тием	30
3.1 Цели и задачи разработки АСУ	30
3.2 Обоснование выбора средств разработки АСУ	31
3.3 Проектирование базы данных для общежития	32
3.3.1 Инфологическое проектирование	32
3.3.2 Логическое проектирование	38
3.3.3 Физическое проектирование	45
3.4 Проектирование базы данных для пользователей АСУ	49
3.4.1 Инфологическое проектирование	49
3.4.2 Логическое проектирование	51

3.4.3 Физическое проектирование	53
4 Программная реализация автоматизированной системы управления общежитием	55
4.1 Описание модулей программы	55
4.2 Алгоритм функционирования программы	57
4.3 Описание пользовательского интерфейса программы	58
4.3.1 Модуль авторизации и окна администратора	58
4.3.2 Главная форма программы	59
4.3.3 Модуль «Проживающие»	60
4.3.4 Модуль «Комнаты»	61
4.3.5 Модуль «Мероприятия»	62
5 Безопасность	63
5.1 Информационная безопасность	63
5.1.1 Разграничение доступа	63
5.1.2 Аутентификация, авторизация и шифрование	64
5.1.3 Безопасность в использовании SQL и резервное копирование	65
5.2 Безопасность и экологичность	66
5.2.1 Безопасность	66
5.2.2 Экологичность	71
5.2.3 Чрезвычайные ситуации	72
Заключение	76
Библиографические ссылки	77
Библиографический список	78
Приложение А (Техническое задание)	80
Приложение Б	91
Приложение В	94
Приложение Г	99
Приложение Д	103

## СОКРАЩЕНИЯ

БД – база данных

СУБД – система управления базой данных

SQL – (Structured Query Language) структурированный язык запросов

ПК – персональный компьютер

ЛВС – локальная вычислительная сеть

АСУ – автоматизированная система управления

ОС – операционная система

## НОРМАТИВНЫЕ ССЫЛКИ

В настоящей бакалаврской работе использованы ссылки на следующие стандарты и нормативные документы:

ГОСТ 2.104-68 ЕСКД Основные надписи

ГОСТ 2.105-95 ЕСКД Общие требования к текстовым документам

ГОСТ 2.106-96 ЕСКД Текстовые документы

ГОСТ 2.111-68 ЕСКД Нормоконтроль

ГОСТ 2.306-68 ЕСКД Обозначение графических материалов и правила нанесения их на чертежах

ГОСТ 2.605-68 ЕСКД Плакаты учебно-технические. Общие технические требования

ГОСТ 19.001-77 ЕСПД Общие положения ГОСТ 19.004-80 ЕСПД Термины и определения

ГОСТ 19.101-77 ЕСПД Виды программ и программных документов ГОСТ 19.102-77 ЕСПД Стадии разработки

ГОСТ 19.103-77 ЕСПД Обозначение программ и программных документов

ГОСТ 19.104-78 ЕСПД Основные надписи

ГОСТ 19.105-78 ЕСПД Общие требования к программным документам.  
ГОСТ 19.106-78 ЕСПД Требования к программным документам, выполненным печатным способом

ГОСТ 19.401-78 ЕСПД Текст программы. Требования к содержанию и оформлению

ГОСТ 19.402-78 ЕСПД Описание программы

ГОСТ 19.502-78 ЕСПД Описание применения. Требования к содержанию и оформлению

ГОСТ 19.504-79 ЕСПД Руководство программиста. Требования к содержанию и оформлению

ГОСТ 19.508-79 ЕСПД Руководство по техническому обслуживанию. Требования к содержанию и оформлению

ГОСТ 24.103-84 Единая система стандартов, автоматизированных систем управления. Автоматизированные системы управления. Основные положения

ГОСТ 24.104-85 Единая система стандартов, автоматизированных систем управления. Автоматизированные системы управления. Общие требования

ГОСТ 24.207-80 Система технической документации на АСУ. Требования к содержанию документов по программному обеспечению

ГОСТ 24.208-80 Система технической документации на АСУ. Требования к содержанию документов стадии «Ввод в эксплуатацию»

ГОСТ 24.209-80 Система технической документации на АСУ. Требования к содержанию документов по организационному обеспечению

ГОСТ 24.210-82 Система технической документации на АСУ. Требования к содержанию документов по функциональной части

ГОСТ 24.301-80 Система технической документации на АСУ. Общие требования к выполнению текстовых документов

ГОСТ 34.201-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначения документов при создании автоматизированных систем

ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания

ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы

ГОСТ 34.603-92 Информационная технология. Комплекс стандартов на автоматизированные системы. Виды испытаний автоматизированных систем



## ВВЕДЕНИЕ

В данной работе представлена разработка автоматизированной системы управления для общежития ФГБОУ ВО «АмГУ».

В условиях современного мира, когда многие процессы подвергаются цифровизации, важность автоматизации управления в общежитиях становится очевидной. Особенно актуальной эта задача стала в свете пандемии COVID-19, которая требует более строгого контроля и прозрачности процессов для обеспечения безопасности студентов и персонала.

Большинство университетов, включая ФГБОУ ВО «АмГУ», имеют общежития и сталкиваются с проблемами управления. Это включает отслеживание студентов, состояние комнат, задачи персонала и другие аспекты. Без должной автоматизации эти проблемы могут привести к недостаточной прозрачности, ошибкам в данных и медленному исполнению административных задач.

Проблема исследования – создание автоматизированной системы управления общежитием, способной улучшить эффективность работы, усилить прозрачность и снизить риск ошибок. В эпоху COVID-19, такой инструмент может также обеспечить более строгий контроль за соблюдением мер безопасности, что в свою очередь улучшит заботу о здоровье студентов и персонала.

Цель исследования – разработать автоматизированную систему управления общежитием для ФГБОУ ВО «АмГУ», которая обеспечит эффективное управление и поддержку общежития, включая отслеживание студентов, состояние комнат, задачи персонала и многое другое.

# 1 АНАЛИЗ ДЕЯТЕЛЬНОСТИ СТУДЕНЧЕСКОГО ГОРОДКА ФГБОУ ВО «АМГУ»

## 1.1 Описание студенческого городка ФГБОУ ВО «АмГУ»

Студенческий городок ФГБОУ ВО «АмГУ» включает в себя комплекс состоящий из 5 общежитий, которые предназначены для проживания студентов на очной или бюджетной форме обучения в период всего срока обучения, за исключением периода новогодних праздников, где от студента потребуется дополнительное соглашение на проживание в этот период времени.

На данный момент времени по актуальным данным из документов расположенных на официальном сайте АмГУ в разделе «Информация о наличии общежитий» всего суммарно доступных мест для проживания студентов составляет 1500 койко-места, где на одного студента рассчитывается площадь 6 кв. м.

В общежитиях ФГБОУ ВО «АмГУ» устанавливаются правила внутреннего распорядка в соответствии с официальным документом утвержденным по указу ректора 10.06.2013 №177-ОД.

В соответствии с правилами внутреннего распорядка, администрация студгородка обеспечивает работу таких процессов в общежитии как:

- заселение, выселение студентов из общежитий;
- обеспечение соблюдения правил проживания в общежитии студентами, а также соблюдение правил по охране труда сотрудников;
- проведение плановых мероприятий по подготовке к действиям в случае пожарной тревоги;
- обеспечение контроля за инвентарным имуществом в общежитии, регулярная инвентаризация имущества, добавление или удаление позиций в журнале инвентаря;
- проведение контроля по оплате студентами за проживание в общежитии в соответствии со сроками оплаты установленными ФГБОУ ВО «АмГУ»;

## 1.2 Функции общежитий

Перечень функций Общежития реализуется соответствующими штатными сотрудниками Общежития, а именно:

- организация рационального использования общежития;
- вселение и дальнейшее проживание в Общежитии обучающихся и работников университета с соблюдением проживающими установленных нормативными документами правил, выселение;
- соблюдение правил регистрации проживающими;
- соблюдение комплекса мер по противопожарному режиму в Общежитии и охране труда;
- осуществление мероприятий по совершенствованию жилищных и культурно-бытовых условий, эстетическому оформлению помещений, благоустройству прилегающей территории;
- обеспечение предоставления документов для регистрации проживающих по месту пребывания;
- содержание помещения студгородка в соответствии с установленными санитарными правилами;
- укомплектование общежития мебелью, оборудованием, постельными принадлежностями и другим инвентарем;
- обеспечение проведения текущего ремонта студгородка, инвентаря, оборудования, содержать в надлежащем порядке закрепленную за студгородком территорию, зеленые насаждения;
- оперативное устранение неисправностей в системах канализации, электроснабжения, водоснабжения студгородка;
- обеспечение предоставления проживающим в студгородке необходимых помещений для самостоятельных занятий, комнат отдыха, бытовых помещений;
- в случае заболевания обучающихся переселять их в другое изолированное помещение по рекомендации лечащего врача;

- обеспечить ежедневный обход всех помещений студгородка с целью выявления недостатков по их эксплуатации и санитарному содержанию и принимать своевременные меры по их устранению;
- производить замену постельного белья не реже одного раза в 10 дней; предоставить проживающим в студгородке право пользоваться бытовой техникой и аппаратурой при соблюдении ими техники безопасности и инструкций по пользованию бытовыми электроприборами;
- содействовать работе студсовета по вопросам улучшения условий проживания, быта и отдыха проживающих;
- принимать меры по реализации предложений проживающих, информировать их о принятых решениях; обеспечивать проживающих необходимым оборудованием, инвентарем, инструментом и материалами для проведения на добровольной основе работ по уборке общежития и закрепленной за студгородком территории;

### **1.3 Организационная структура студенческого городка**

Численный состав работников студенческого городка и перечень должностей определяются штатным расписанием университета. Численность лиц проживающих в общежитиях регламентируется вместимостью жилых и иных помещений и установленными санитарно-техническими нормами площадей на одного проживающего. (Статья 105 «Жилищного кодекса РФ»). Структура студенческого городка включает: дирекцию студенческого городка, администрацию и обслуживающий персонал общежитий. Их обязанности, права и ответственность определяются должностными инструкциями, настоящим Положением, Положением об общежитии и иными локальными нормативными актами, регламентирующими деятельность студенческого городка. На рисунке **1** представлена организационная структура студенческого городка.

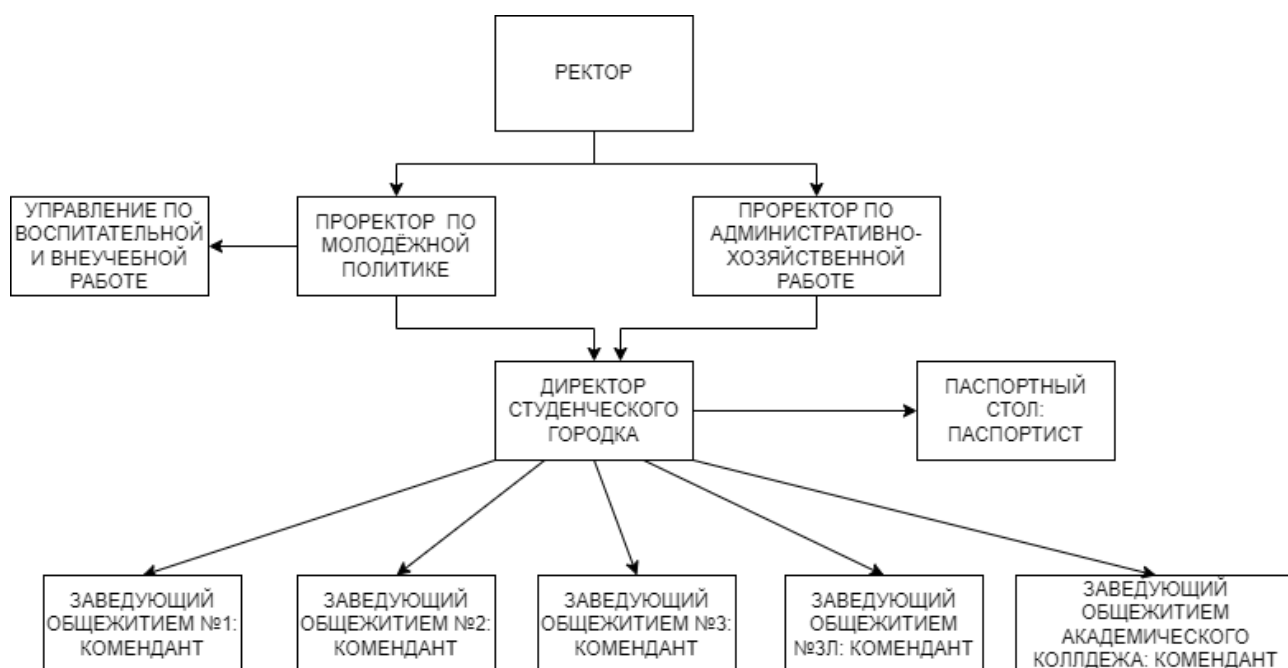


Рисунок 1 – Организационная структура студенческого городка

Директор студенческого городка - центральная фигура в этой структуре. Он имеет прямые связи с несколькими ключевыми лицами и подразделениями:

- заместитель декана по воспитательной работе;
- паспортист в паспортном столе;
- общежития (общежитие 1, общежитие 2, общежитие 3, общежитие 3л, общежитие академического колледжа);
- начальник охраны;
- заместитель декана по воспитательной работе и паспортист в паспортном столе - взаимодействуют напрямую с директором студенческого городка;
- общежития - каждое общежитие имеет прямую связь с директором студенческого городка и своим комендантом;
- начальник охраны - под его управлением находятся охранники всех общежитий. Он также имеет прямую связь с директором студенческого городка;
- охранники общежитий - каждый охранник связан с начальником охраны;

– коменданты общежитий - каждый комендант связан с соответствующим общежитием;

#### 1.4 Описание потоков данных и бизнес-процессов в общежитиях

Так как в данной работе прodelывается работа по созданию ИС для автоматизации процессов управления внутри общежития, то и рассматриваться будут только соответствующие процессы, которые не будут пересекаться с другими внешними функциями общежития, которые не относятся к процессу управления.

Одними из главных процессов общежития являются:

*Процесс заселения студентов в общежитие, структура изображена на рисунке 2.*

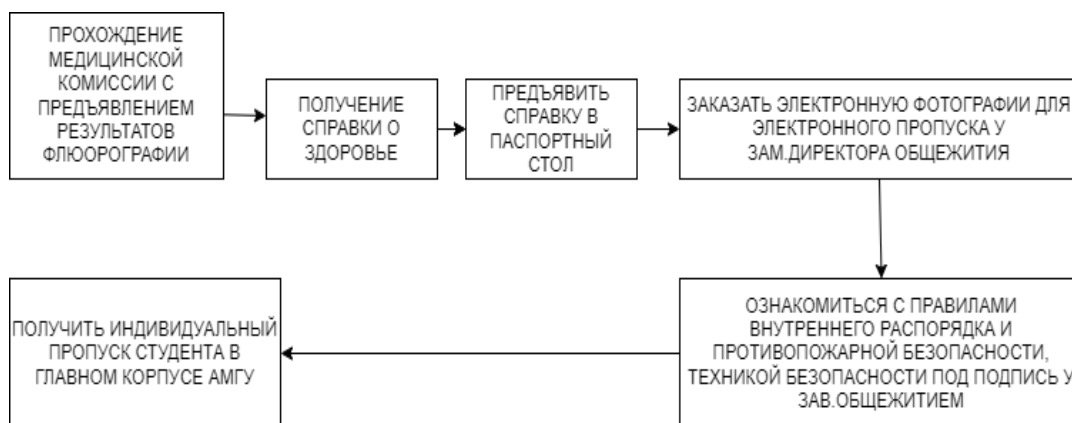


Рисунок 2 – процесс заселения студента в общежитие

*Процесс выселения студента из общежития, структура изображена на рисунке 3.*

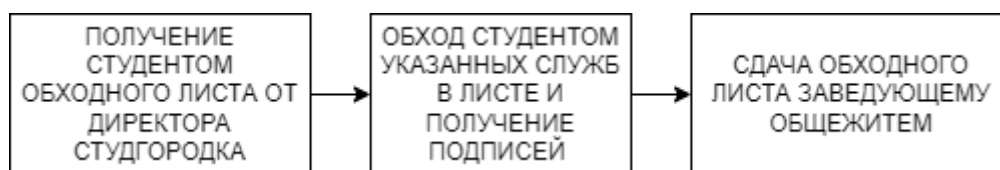


Рисунок 3 – процесс выселения студента в общежитие

*Процесс надзора по соблюдению проживающими правил проживания в общежитии, периодического информирования о правилах проживания, структура изображена на рисунке 4.*

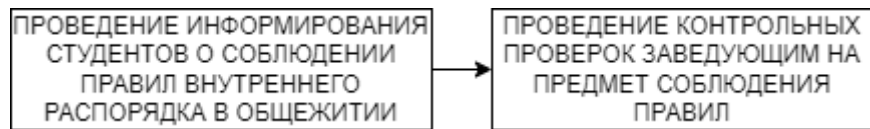


Рисунок 4 – процесс надзора по соблюдению внутренних правил общежития

*Процесс соблюдения комплекса мер по противопожарному режиму и охране труда сотрудников, структура изображена на рисунке 5.*

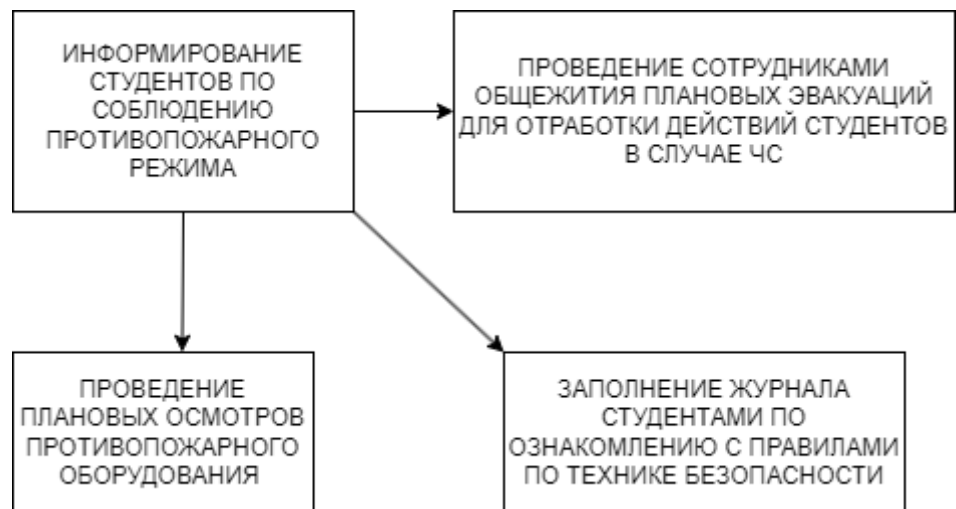


Рисунок 5 – процесс надзора по соблюдению внутренних правил общежития

*Процесс инвентаризации, периодическая проверка инвентаря, добавление или удаление позиций, структура изображена на рисунке 6.*

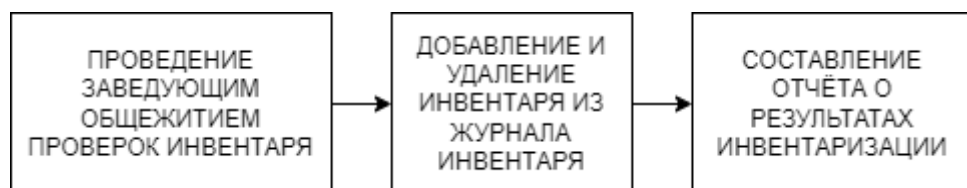


Рисунок 6 – процесс надзора по соблюдению внутренних правил общежития

В соответствии с описанными бизнес-процессами далее будут представлены основные модели потоков данных, протекающих в общежитиях студенческого городка ФГБОУ ВО «АмГУ». Моделирование будет происходить при помощи такого инструмента как AllFusion Process Modeller (PM).

AllFusion PM предлагает средство для сбора всей необходимой информации о работе предприятия и графического изображения этой информации в виде целостной и непротиворечивой модели. Простота и наглядность моделей Process Modeler упрощает взаимопонимание между всеми участниками бизнес-процессов: от руководителей до исполнителей.

На рисунке 7 представлена диаграмма композиции процесса заселения студентов в общежитие.

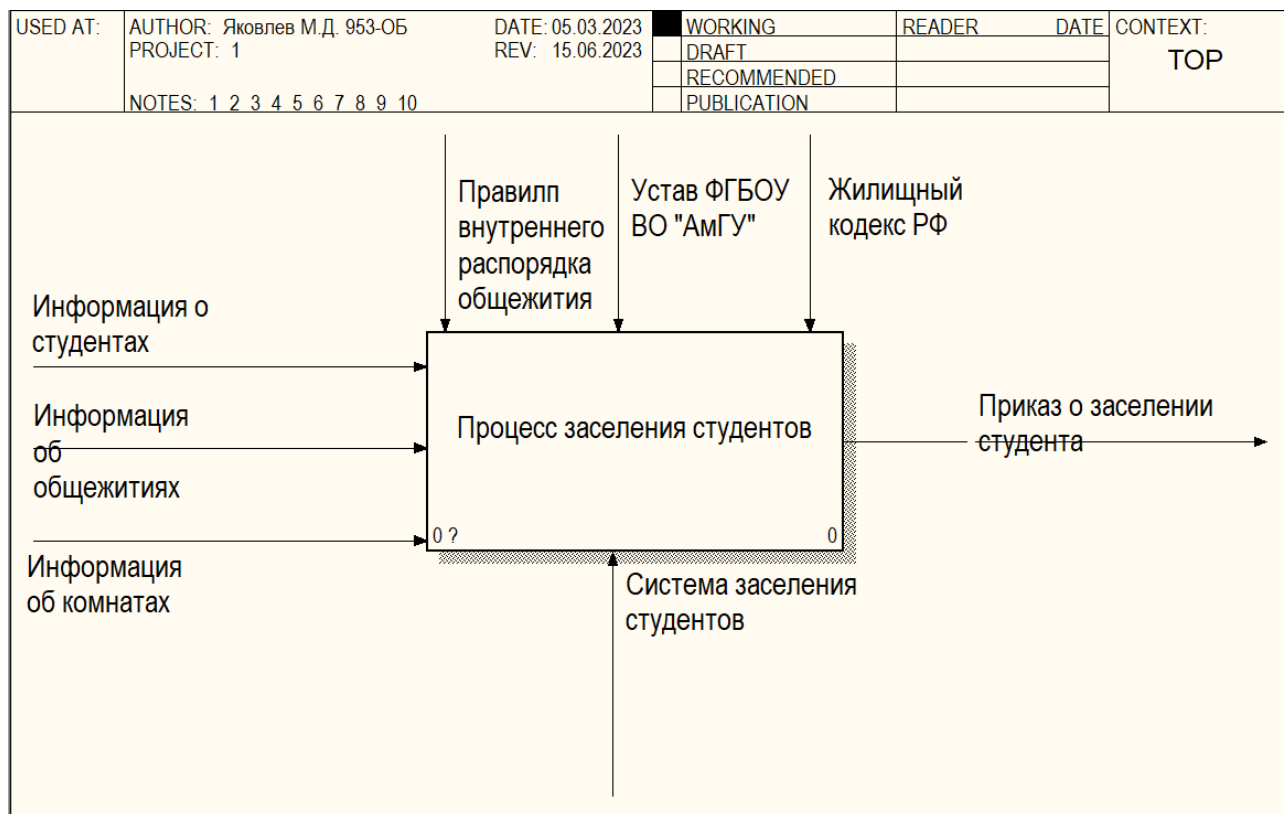


Рисунок 7 – Композиция процесса заселения студентов

На рисунке 8 представлена диаграмма декомпозиции процесса заселения студентов.



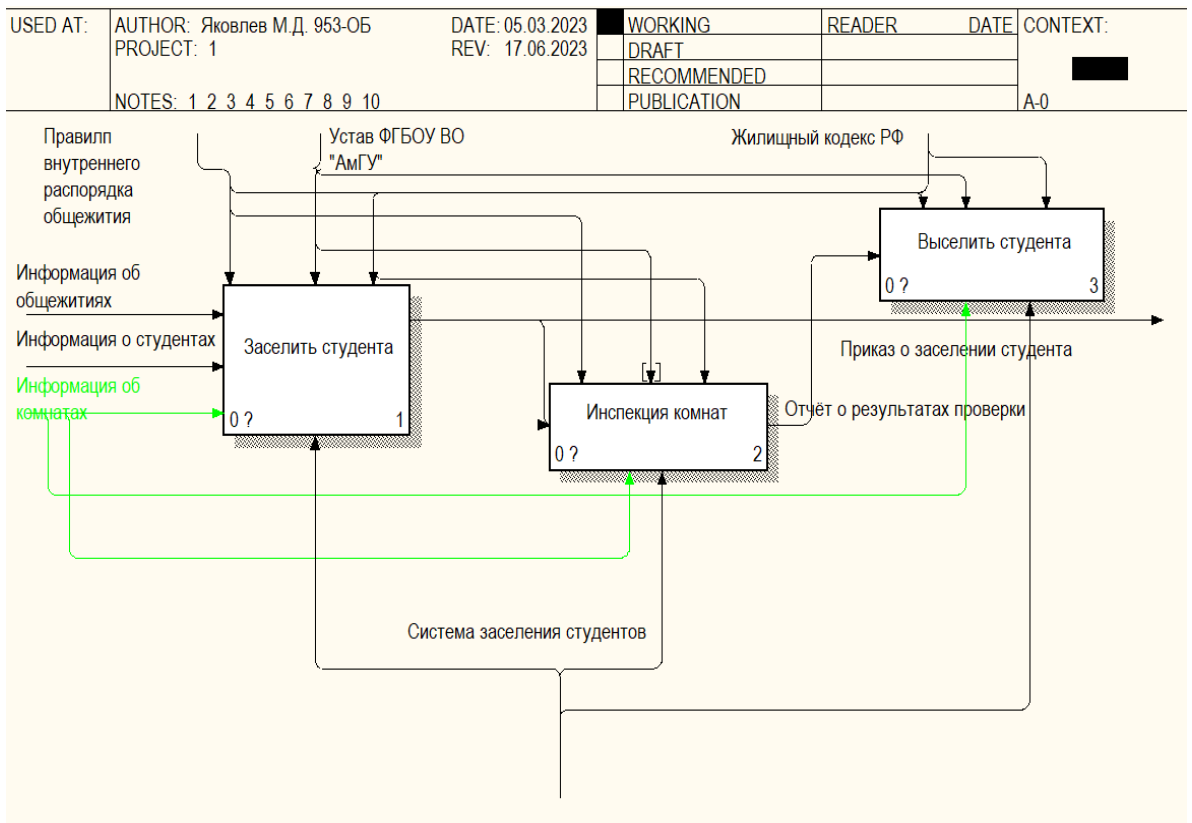


Рисунок 8 – Диаграмма декомпозиции процесса заселения

На рисунке 9 представлена диаграмма декомпозиции процесса «Заселить студента».

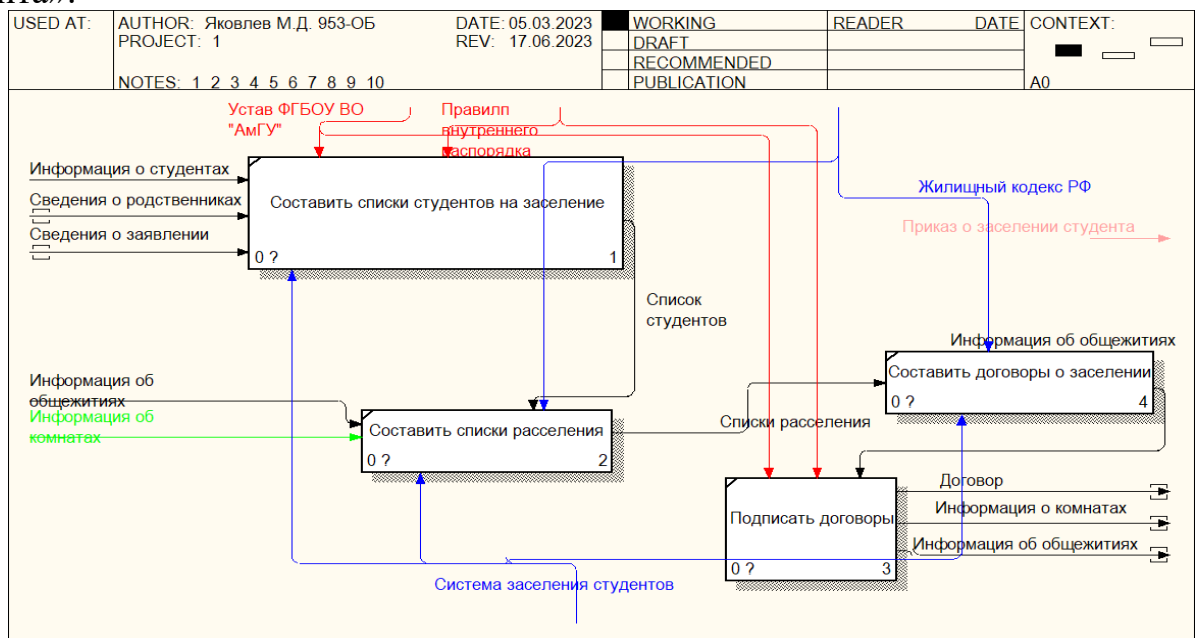


Рисунок 9 – Декомпозиция процесса «Заселить студента»

На рисунке 10 представлена диаграмма декомпозиции процесса «Инспекция комнат».

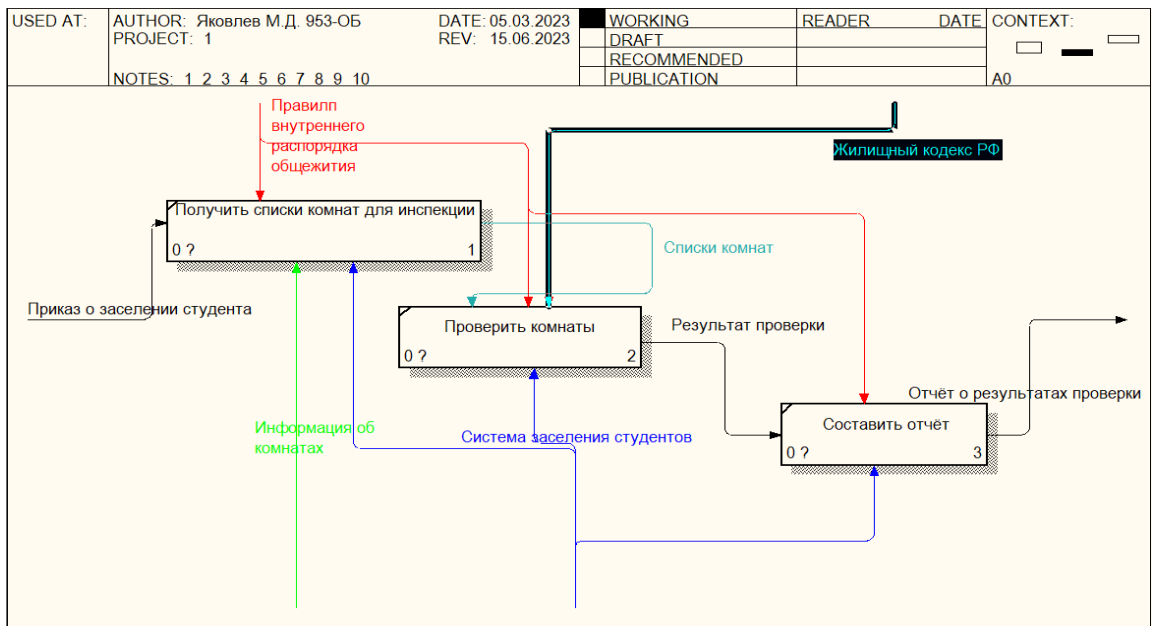


Рисунок 10 – Диаграмма декомпозиции процесса «Инспекция комнат»

На рисунке 11 представлена диаграмма декомпозиции процесса «Выселить студента».

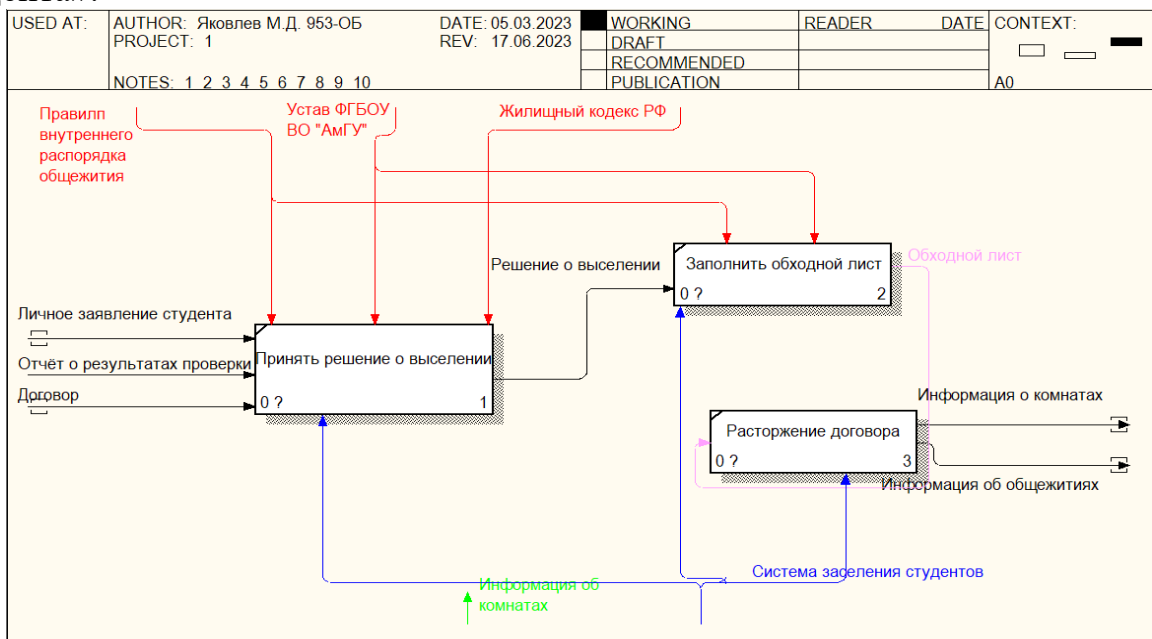


Рисунок 11 – Диаграмма декомпозиции процесса «Выселить студента»

### 1.5 Обзор и анализ существующих проектных решений, выявление достоинств, недостатков

Первая программа называется АИС «Общежитие» (рисунок 12). Официальный сайт <http://www.ca-c.ru/>. Разработчики данной программы компания ООО «Спецавтоматика-Спектр». Программа имеет на текущий момент версию 0.2.0.0 и предназначена для хранения и обработки информации о студентах, аспирантах,

сотрудниках ВУЗа, проживающих в общежитиях. АИС «Общежитие» содержит средства поддержки, контроля и анализа процессов, связанных с заселением, учетом лиц проживающих в общежитиях ВУЗа. В АИС «Общежитие» предусмотрена система аутентификации, ведется журнал изменений позволяющий отследить любое изменение информационных полей пользователем системы. На сайте программа доступна в демо-режиме. Программа является платной. Стоимость одной лицензии 6999 рублей. Работает с СУБД MySQL.

Вторая программа называется «Электронное общежитие – сетевая версия» (рисунок 13). Официальный сайт <http://pbprog.ru>. Разработчики данной программы Батищев Павел Сергеевич. Программа имеет на текущий момент версию 1.0.12.56. Программа предназначена для автоматизации работы комендантов и воспитателей общежитий и может быть использована как в общежитиях учебных заведений начального, среднего, высшего профессионального образования, повышения квалификации, так и в общежитиях предприятий, организаций. Для работы с программой необходим установленный Microsoft SQL Server. Стоимость программы 14950 рублей.

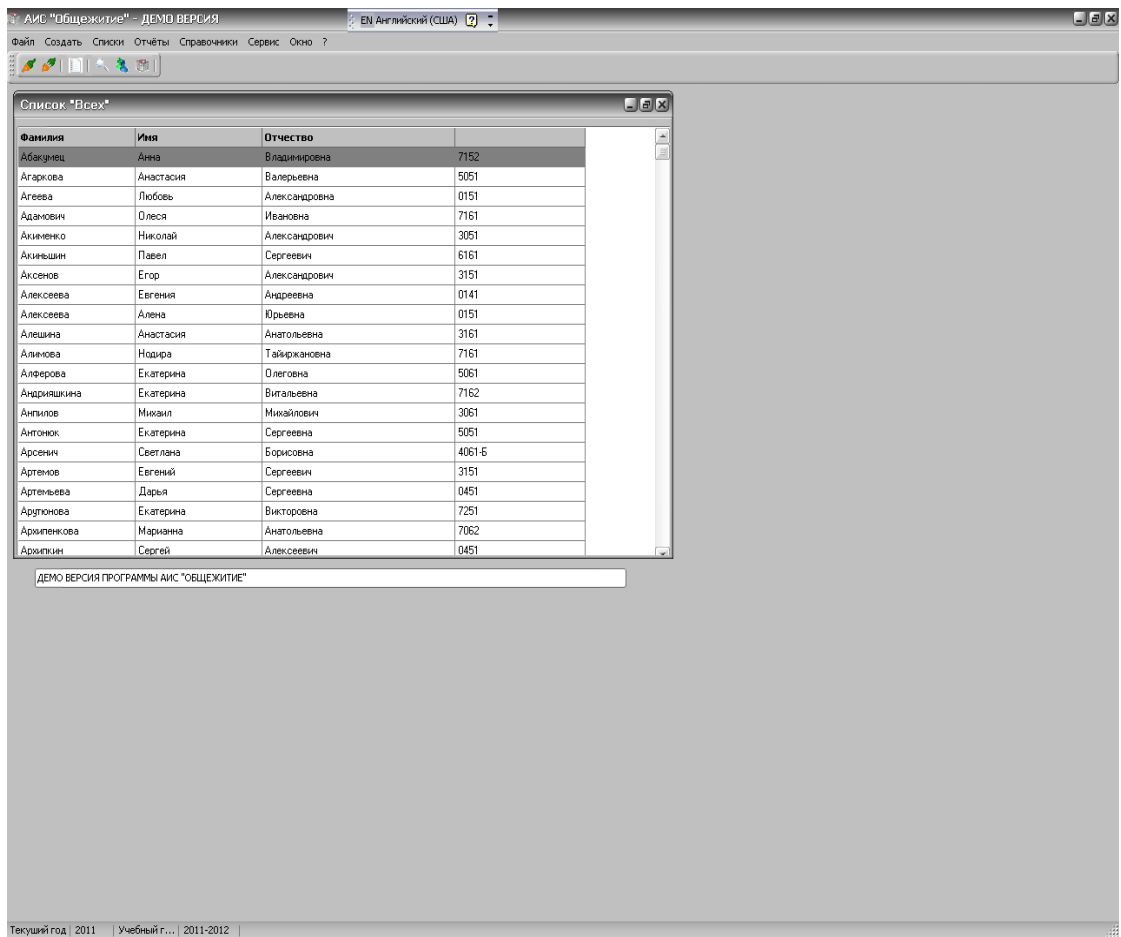


Рисунок 12 – Программа АИС «Общежитие»

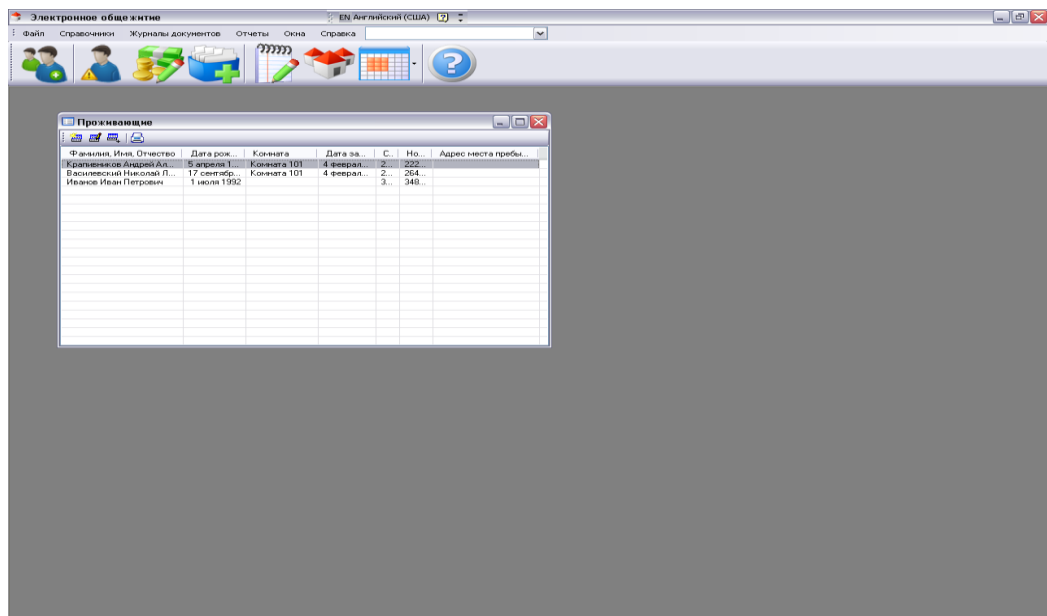


Рисунок 13 – Программа «Электронное общежитие – сетевая версия»

Третья программа называется БИТ.Общежитие 8. Представляет собой 1С приложение для управления общежитием. Сайт <https://blagoveshchensk.1cbit.ru/>.

Данное приложение имеет следующие функции:

- гибкое планирование индивидуальной структуры общежития;
- контроль за проживающими и оперативный учет проживающих в общежитии (заезд, перемещение, выезд);
- учет материальных ценностей, числящихся за общежитием или закрепленных за проживающими;
- учет лиц, дополнительно проживающих на жилплощади совместно с лицом, заключившим договор с общежитием;
- учет предоставления дополнительных услуг (прачечная, телефон и пр.) в общежитии;
- учет наличия койко-мест и их характеристик;
- отчетность по численности проживающих и дополнительно проживающих, взаиморасчетам и задолженности проживающих, наличию свободных койко-мест, материальным средствам;

Цена лицензии на данный момент для одного рабочего места составляет 5000 рублей.

Акт об оказании услуг 000000001 от 22.02.2017 13:57:17

Провести и закрыть | Провести | Акт об оказании услуг | Все действия ?

Номер: 000000001 от: 22.02.2017 13:57:17 | Общежитие: Общежитие № 2

Плательщик: ООО "Мистер ИКС" | Начисление за период с: .. по: .. (+)

Содержание: | Валюта: RUB: НДС включен в сумму

Доп. услуги (4)

N	Проживающий	Номенклатура	Цена	Сумма	Ставка НДС	Всего	Содержание
	Договор	Ед. изм.	Количество		Сумма НДС		
1	Маркелов Александр Договор 624	Прачечная шт	1,00	50,00	18%	7,63	50,00 Прачечная
2	Малевич Казимир	Прачечная шт	1,00	50,00	18%	7,63	50,00 Прачечная
3	Степанова Юлия Ива...	Прачечная шт	1,00	50,00	18%	7,63	50,00 Прачечная
4	Степанова Юлия Ива...	Прачечная шт	1,00	50,00	18%	7,63	50,00 Прачечная

Всего: 200,00 RUB НДС (в т.ч.): 30,52 RUB

Ответственный: Администратор

Комментарий:

Рисунок 14 – Составление отчёта в БИТ. Общежитие

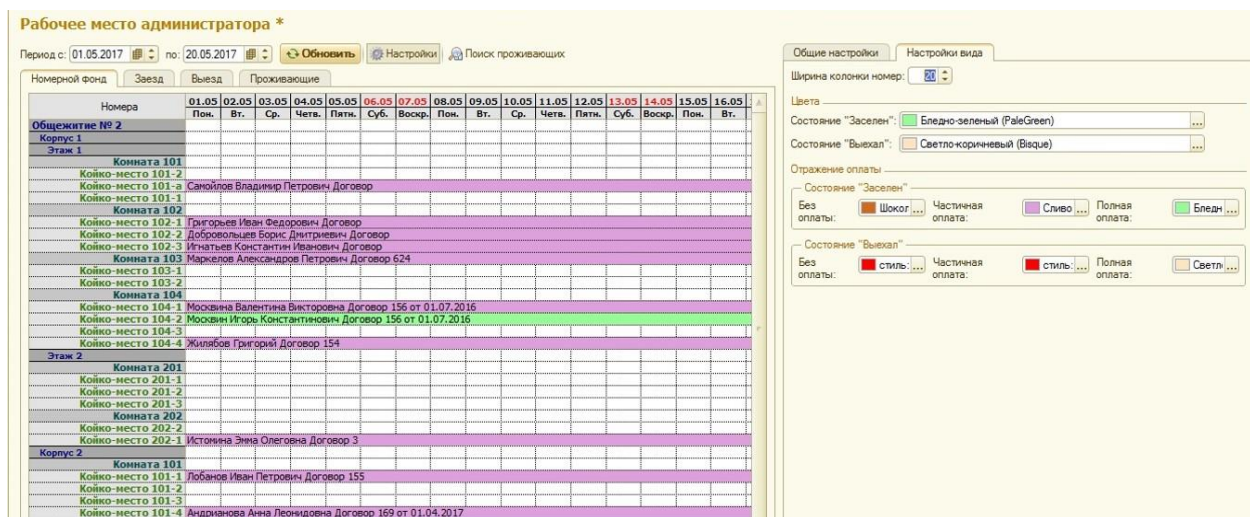


Рисунок 15 – Рабочее место администратора в БИТ. Общежитие

Проанализировав существующие аналоги, можно прийти к заключению, что наиболее подходящим вариантом может являться БИТ. Общежитие 8, так как имеет хороший функционал, а также имеет удобный и простой интерфейс. Имеются дополнительные заранее созданные шаблоны для заполнения отчётов. Однако высокая цена за ПО является препятствием для его периодической закупки.

Остальные ранее рассмотренные ПО можно отнести к неудовлетворительным по условиям. Основные проблемы:

- программы «Электронное общежитие – сетевая версия» и «Общежитие» являются платными программами;
- программы «Электронное общежитие – сетевая версия» и «Общежитие» являются устаревшими. Данное ПО не обновлялось, что означает что они могут не соответствовать как нормам безопасности, так и нормам оформления;

Так как ПО давно не обновлялось, то и платформа разработки приложений на данный момент времени уже не является актуальной, так как SQL Server имеет обновление от 2022 года.

## 1.6 Обоснование необходимости разработки АСУ

Обоснование необходимости разработки автоматизированной системы управления (АСУ) общежитием для ФГБОУ ВО "АмГУ" на базе PostgreSQL основывается на нескольких ключевых аспектах:

- оптимизация управленческих процессов: АСУ позволяет оптимизировать и улучшить процессы управления общежитием, что в свою очередь повышает эффективность работы персонала и уменьшает время, затрачиваемое на выполнение административных задач;

- повышение комфорта и безопасности жильцов: система автоматизации улучшит учет студентов и их поселение, предоставит быстрый и удобный доступ к информации, что в свою очередь повышает комфорт проживания. Кроме того, автоматизация поможет уменьшить риск ошибок при ведении учета и обработке данных, что важно для обеспечения безопасности жильцов;

- пандемия COVID-19 и ограничения на контактное общение: в условиях ограничений на контактное общение, автоматизированные системы управления становятся особенно важными. Они позволяют сократить контакты между людьми и увеличить безопасность обитателей общежития;

- экономическая выгода: АСУ позволит снизить затраты на приобретение и поддержку программного обеспечения в будущем, что делает проект экономически выгодным;

В целом, разработка АСУ для общежития ФГБОУ ВО "АмГУ" является актуальной и необходимой задачей, способной значительно улучшить процесс управления общежитием и повысить качество проживания для студентов.

## 2 ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ ПОСТРОЕНИЯ И ВЫБОРА ПЛАТФОРМЫ РЕАЛИЗАЦИИ АСУ

### 2.1 Выбор архитектуры построения АСУ

Основными аспектами при выборе архитектуры построения ИС являются быстродействие, надежность, масштабируемость и безопасность.

В настоящее время наиболее распространенными архитектурами являются:

- файл-сервер;
- клиент-сервер;
- многоуровневая архитектура;

Файл-серверная архитектура подразумевает под собой то, что сервер возлагает на себя лишь функцию хранения данных, а обработка производится на клиентских машинах. Это означает, что данные необходимо передавать по сети, что приведет к сильной загрузке сетевого трафика. А это в свою очередь приведет к снижению производительности при увеличении числа пользователей. Также при реализации архитектуры файл-сервер, проблема целостности, согласованности и одновременного доступа к данным решается децентрализованно: данные хранятся на сервере, а обрабатываются на клиенте. Вследствие этого снижается надежность приложения. Еще одним недостатком являются высокие затраты на модернизацию и сопровождение сервисов бизнес - логики на каждой клиентской рабочей станции. Однако данная архитектура обладает и рядом преимуществ, таких как низкая стоимость разработки, высокая скорость разработки и невысокая стоимость обновления и изменения программного обеспечения.

Архитектура клиент-сервер лишена недостатков вышеописанной архитектуры, т.к. сервер баз данных не только обеспечивает доступ к общим данным, но и выполняет их обработку. Клиент посылает на сервер запросы, на языке «понятном» серверу, а он в свою очередь обрабатывает запрос, контролируя при этом целостность и согласованность данных, и возвращает на клиент результат обработанного запроса. В результате нагрузка на сеть снижается: клиенту больше не



нужно обрабатывать промежуточные данные. Хранение и обработка производится централизованно, поэтому данная архитектура надежнее архитектуры файл-сервер. К недостаткам клиент-серверной архитектуры относятся, во-первых, достаточная сложность разработки системы из-за необходимости исполнять бизнес-логику и обеспечивать интерфейс с пользователем в одной программе и высокие требования к рабочим станциям по той же причине.

Следующей ступенью развития архитектур ИС стала многоуровневая архитектура, в которой бизнес-логика выполняется на сервере приложений. Многоуровневая архитектура обладает следующими достоинствами:

- масштабируемость;
- конфигурируемость - изолированность уровней друг от друга позволяет быстро и простыми средствами переконфигурировать систему при возникновении сбоев или при плановом обслуживании на одном из уровней;
- высокая безопасность;
- высокая надёжность;
- низкие требования к скорости канала (сети) между терминалами и сервером приложений;
- низкие требования к производительности и техническим характеристикам терминалов, как следствие снижение их стоимости;

Однако, несмотря на неоспоримые достоинства, данная система не получила распространения, по следующим причинам:

- сложность разработки систем на основе многоуровневой архитектуры, т. к. очень сложно «состыковать» различные модули, особенно если они написаны разными группами. А изменение в одном модуле, как правило, вызывает лавинообразные изменения в остальных, и с этой точки зрения даже простую систему, основанную на многоуровневой архитектуре, будет сложнее выполнить в 2 раза;
- высокие требования к производительности серверов приложений и сервера базы данных, а, значит, и высокая стоимость серверного оборудования;

- высокие требования к скорости канала (сети) между сервером базы данных и серверами приложений;
- высокая сложность администрирования;

Рассмотрев все достоинства и недостатки каждой из архитектур, для реализации автоматизированной системы управления общежитием выбираем архитектуру клиент-сервер. Данная архитектура позволяет оптимально распределить работу между клиентскими и серверными частями системы: приложение, работающее на рабочей станции, не читает записи базы данных «напрямую», а посылает запросы на сервер, где они последовательно обрабатываются, а результаты обработки отсылаются на рабочую станцию. А это существенно сокращает информационные потоки в ЛВС.

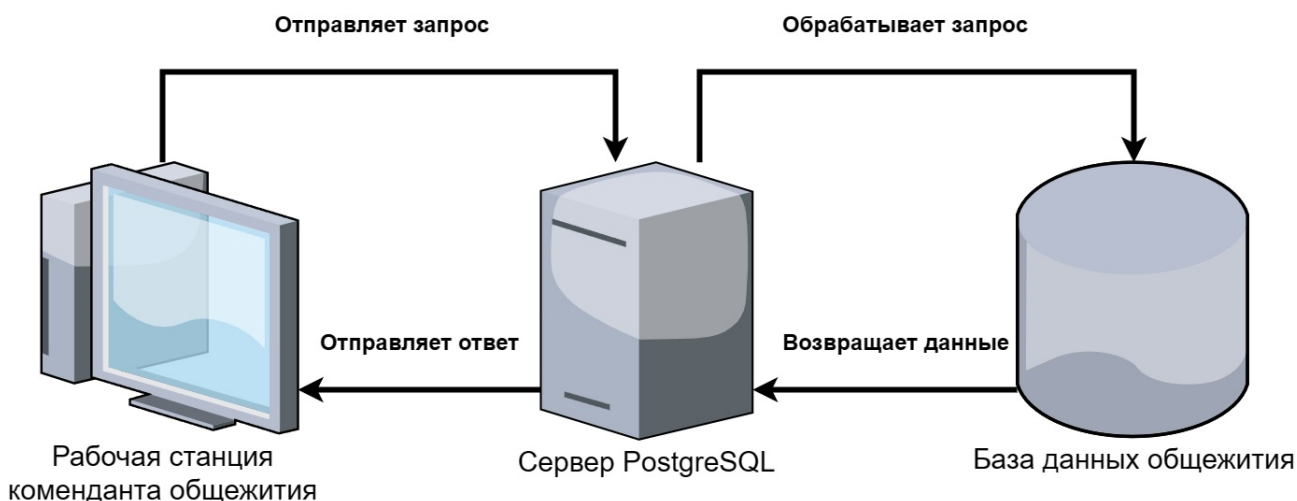


Рисунок 16 – Структура архитектуры «Клиент-Сервер»

## 2.2 Технические характеристики ПК-клиента АСУ

Для успешного функционирования автоматизированной системы управления (АСУ) общежитием, технические характеристики как клиентских персональных компьютеров (ПК), так и сервера играют ключевую роль. Это обеспечивает надежную и стабильную работу системы, эффективное выполнение задач и комфортную работу пользователей.

Минимальные технические характеристики ПК-клиента могут включать следующие параметры:

- *процессор (CPU)*: Процессор с частотой не менее 2.0 ГГц. Это обеспечивает достаточную вычислительную мощь для выполнения операций, связанных с использованием АСУ;
- *оперативная память (RAM)*: минимум 4 ГБ оперативной памяти, чтобы обеспечить достаточно пространства для многозадачности и эффективной работы приложений;
- *жесткий диск (HDD) или твердотельный диск (SSD)*: Объемом не менее 128 ГБ для хранения данных и установки необходимого программного обеспечения;
- *видеокарта*: Основная интегрированная видеокарта с поддержкой разрешения не менее 1024x768 пикселей, чтобы обеспечить четкое отображение интерфейса приложений;
- *операционная система*: Windows 10 или новее для полной совместимости с приложением, написанным на C# и WPF формах;
- *сетевое подключение*: Ethernet (1Gbit/s), Wi-Fi (IEEE 802.11b/g/n/ac);
- *периферийные устройства*: клавиатура и мышь для ввода данных, монитор для отображения информации;

Технические характеристики сервера могут включать следующие параметры:

- *процессор (CPU)*: Мощный процессор с частотой не менее 2.5 ГГц и несколькими ядрами для обработки большого количества запросов от клиентов;
- *оперативная память (RAM)*: минимум 8 ГБ оперативной памяти для эффективного обслуживания всех клиентских подключений и обработки запросов;
- *жесткий диск (HDD) или твердотельный диск (SSD)*: минимум 500 ГБ для хранения большого количества данных, учетных записей пользователей и истории операций;
- *операционная система*: Windows Server, Linux или другая серверная операционная система, совместимая с PostgreSQL;
- *сетевое подключение*: Ethernet (1Gbit/s), Wi-Fi (IEEE 802.11b/g/n/ac);

– *программное обеспечение*: Наличие установленного и настроенного PostgreSQL;

### **2.3 Обеспечивающие и функциональные подсистемы АСУ**

Автоматизированная система управления (АСУ) включает в себя обеспечивающие и функциональные подсистемы. Обеспечивающие подсистемы обычно создают среду для функционирования функциональных подсистем, которые выполняют непосредственные задачи по управлению и контролю.

Обеспечивающие подсистемы включают:

– подсистема управления данными. Эта подсистема управляет базами данных и обеспечивает доступ к данным. Она также обеспечивает сохранность данных, их целостность и конфиденциальность;

– подсистема управления безопасностью. Эта подсистема управляет доступом к системе и данным, реализуя механизмы аутентификации, авторизации и аудита;

– подсистема сетевого взаимодействия. Эта подсистема обеспечивает подключение к сети и передачу данных между клиентами и сервером;

– подсистема обеспечения работы с пользовательским интерфейсом. Она предоставляет средства для взаимодействия пользователей с системой;

Функциональные подсистемы могут включать:

– подсистема управления проживанием студентов. Эта подсистема управляет информацией о проживающих студентах, их присутствии, оплате и других связанных вопросах;

– подсистема управления персоналом. Эта подсистема управляет информацией о сотрудниках общежития, их расписанием, обязанностями;

– подсистема управления обслуживанием. Эта подсистема управляет информацией об обслуживании, включая уборку, ремонт и обслуживание общежития;

– подсистема управления финансами. Эта подсистема управляет финансовой информацией, включая платежи, счета, финансовые отчеты;

Все эти подсистемы вместе обеспечивают эффективную работу автоматизированной системы управления общежитием.

На рисунке 17 представлена структура обеспечивающих и функциональных подсистем автоматизированной системы управления общежитием для ФГБОУ ВО «АмГУ».

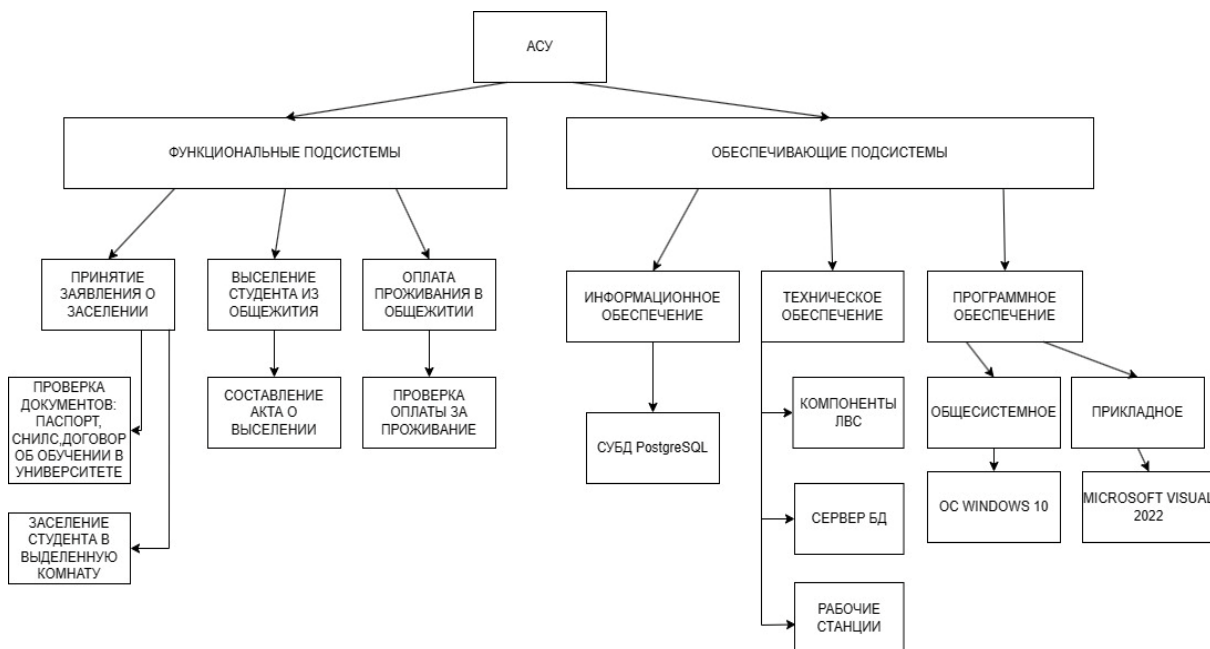


Рисунок 18 – Структура обеспечивающих и функциональных подсистем АСУ

## 3 ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ОБЩЕЖИТИЕМ

### 3.1 Цели и задачи разработки АСУ

Целью создания автоматизированной системы управления (АСУ) для общежития ФГБОУ ВО «АмГУ» является улучшение управления общежитием, повышение комфорта и безопасности жильцов, а также усовершенствование процессов выполнения административных задач. В современных условиях и особенно в контексте ограничений, связанных с COVID-19, применение АСУ может стать важным инструментом обеспечения безопасности и управления общежитиями.

Специфические цели создания системы включают:

- упрощение процесса учета студентов и их поселения в общежитии;
- увеличение прозрачности и эффективности управления общежитием;
- обеспечение быстрого и удобного доступа к информации о студентах, сотрудниках, комнатах и состоянии общежития;
- сокращение времени на выполнение административных задач и повышение качества обслуживания студентов;
- минимизация риска ошибок при ведении учета и обработке данных;
- создание возможности для автоматического формирования отчетов и статистики по управлению общежитием;
- расширение функциональности и улучшение производительности работы персонала, связанного с управлением общежитием;
- для достижения этих целей, необходимо решить следующие задачи:
- выбор средств и среды разработки;
- разработка базы данных для хранения информации о студентах, комнатах, записях в журнале, сотрудниках;
- разработка пользовательского интерфейса, позволяющего быстро и удобно добавлять, редактировать и просматривать информацию;

- реализация проекта с помощью программных средств;

Система будет разработана на базе PostgreSQL, что не только позволит снизить затраты на приобретение и поддержку ПО, но и обеспечит совместимость с отечественными операционными системами, такими как Astra Linux и другими Linux.

### **3.2 Обоснование выбора средств разработки АСУ**

При выборе средств разработки автоматизированной системы управления общежитием ФГБОУ ВО «АмГУ» были учтены ключевые требования к функциональности, надежности, совместимости и стоимости.

В качестве основы для разработки была выбрана база данных PostgreSQL. Этот выбор обусловлен рядом преимуществ данной системы:

- *открытый исходный код*: PostgreSQL — это система управления базами данных с открытым исходным кодом, что позволяет свободно использовать и модифицировать ее, а также значительно снижает затраты на приобретение и поддержку ПО;

- *производительность и надежность*: PostgreSQL известна своей высокой производительностью, масштабируемостью и надежностью, что гарантирует стабильную работу системы даже при больших объемах данных и высокой нагрузке;

- *безопасность*: PostgreSQL предлагает мощные механизмы безопасности, включая поддержку SSL для шифрования данных, а также различные методы аутентификации;

- *совместимость*: PostgreSQL совместим с большим количеством операционных систем, включая Windows, Linux, UNIX и macOS, что обеспечивает гибкость в выборе платформы для размещения системы. Кроме того, PostgreSQL совместим с отечественными операционными системами, такими как Astra Linux;

- *поддержка SQL и расширяемость*: PostgreSQL полностью поддерживает SQL и предлагает множество расширений, которые могут быть использованы для дополнительного расширения функциональности системы;

Средой разработки для этого проекта будет C# на платформе Windows Presentation Foundation (WPF).

- C#: это объектно-ориентированный язык программирования, предлагающий широкий спектр возможностей и функций. C# является одним из наиболее популярных и востребованных языков программирования, благодаря его мощности, гибкости и поддержке со стороны Microsoft;

- WPF (Windows Presentation Foundation): это библиотека графического пользовательского интерфейса для создания настольных приложений на платформе Windows. WPF предлагает большую гибкость и контроль над визуальным представлением приложения, что позволяет создать интуитивно понятный и привлекательный пользовательский интерфейс;

Эти выборы вместе обеспечивают набор мощных и гибких инструментов для разработки автоматизированной системы управления, которая будет соответствовать всем требованиям и потребностям университета.

### **3.3 Проектирование базы данных**

#### **3.3.1 Инфологическое проектирование**

Инфологическое проектирование — это первый этап проектирования базы данных, который включает определение структуры данных, типов данных и связей между ними.

Инфологическое проектирование позволяет представить базу данных в виде набора связанных сущностей, описывающих основные объекты и явления в предметной области. Сущности характеризуются набором атрибутов, каждый из которых имеет определенный тип данных.

Данный этап проектирования базы данных предполагает глубокое понимание предметной области и задач, которые будут решаться с помощью разрабатываемой системы. Важными компонентами инфологического проектирования являются правильное определение атрибутов каждой сущности, выбор первичных и внешних ключей для обеспечения связей между сущностями и гарантии целостности данных.



Для каждой из выделенных сущностей были определены основные атрибуты и типы данных, использованные для их хранения, а также внешние ключи для связи с другими сущностями. Это позволяет обеспечить гибкость и масштабируемость базы данных, а также облегчает процесс добавления новых данных и изменения существующих.

Построение инфологической модели базы данных является ключевым этапом проектирования, определяющим структуру и организацию данных в системе. Корректное и грамотное инфологическое проектирование является гарантией надежности и эффективности функционирования системы в будущем.

При проектировании базы данных автоматизированной системы управления общежитием ФГБОУ ВО «АмГУ» выделено 11 основных сущностей: должности\_сотрудников, сотрудники, комнаты, студенты, электроустановки, студент\_электроустановки, оплата, заселение\_выселение, мероприятия, запросы\_на\_вызов\_сотрудников, имущество\_комнат. Подробное описание каждой сущности приведено в таблице 1.

Таблица 1 – Сущности

Название	Описание
должности_сотрудников	Содержит информацию о различных должностях сотрудников общежития
сотрудники	Содержит информацию о сотрудниках общежития, включая их имя, контактные данные и должность
комнаты	Содержит информацию о всех комнатах в общежитии, включая их номер, этаж, вместимость и стоимость проживания
студенты	Содержит информацию о студентах, проживающих в общежитии, включая их имя, контактные данные и номер комнаты
электроустановки	Содержит информацию об электроустановках, которые используют студенты
студент_электроустановки	Связывает студентов и их электроустановки
оплата	Содержит информацию об оплате за проживание студентами
заселение_выселение	Содержит информацию о заселении и выселении студентов
мероприятия	Содержит информацию о мероприятиях, проводимых в общежитии
запросы_на_вызов_сотрудников	Содержит информацию о запросах на вызов сотрудников общежития для устранения различных проблем
имущество_комнат	Содержит информацию об имуществе комнат

Каждая из этих сущностей содержит ряд, которые предназначены для хранения различных видов информации. При проектировании этих таблиц были учтены необходимые требования к целостности данных и соблюдены принципы

нормализации для обеспечения эффективности и надежности базы данных. Информация об атрибутах представлена в таблице 2-12.

Таблица 2 – Атрибуты сущности «должности\_сотрудников»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор должности	>0	Число	1
название_должности	Название должности	-	Строка	"Директор"

Таблица 3 – Атрибуты сущности «сотрудники»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор сотрудника	>0	Число	1
имя	Имя сотрудника	-	Строка	"Алексей"
фамилия	Фамилия сотрудника	-	Строка	"Иванов"
отчество	Отчество сотрудника	-	Строка	"Михайлович"
номер_телефона	Номер телефона сотрудника	-	Строка	"+7 900 000 00 00"
электронная_почта	Электронная почта сотрудника	-	Строка	"example@example.com"
id_должности	Идентификатор должности	>0	Число	1

Таблица 4 – Атрибуты сущности «комнаты»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор комнаты	>0	Число	101
номер_комнаты	Номер комнаты	-	Строка	"101a"
этаж	Этаж, на котором расположена комната	>0	Число	3
вместимость	Вместимость комнаты	>0	Число	2
стоимость_проживания	Стоимость проживания в комнате	>=0	Десятичное число	2500.00
свободные_места	Количество свободных мест в комнате	>=0	Число	1

Таблица 5 – Атрибуты сущности «студенты»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор студента	>0	Число	1
имя	Имя студента	-	Строка	"Иван"
фамилия	Фамилия студента	-	Строка	"Иванов"
отчество	Отчество студента	-	Строка	"Иванович"

Продолжение таблицы 5

дата рождения	Дата рождения студента	-	Дата	"2000-01-01"
номер телефона	Номер телефона студента	-	Строка	"+7 900 000 00 00"
электронная почта	Электронная почта студента	-	Строка	"example@example.com"
группа	Группа, в которой учится студент	-	Строка	"ИВТ-101"
id комнаты	Идентификатор комнаты, в которой проживает студент	>0	Число	101

Таблица 6 – Атрибуты сущности «электроустановки»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
id	Идентификатор электроустановки	>0	Число	1
тип	Тип электроустановки	-	Строка	"Фен"
стоимость_использования	Стоимость использования электроустановки	>=0	Десятичное число	150.00
id студента	Идентификатор студента, использующего установку	>0	Число	1

Таблица 7 – Атрибуты сущности «оплата»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
id	Идентификатор оплаты	>0	Число	1
id студента	Идентификатор студента, производящего оплату	>0	Число	1
сумма оплаты	Сумма оплаты	>=0	Десятичное число	2500.00
дата оплаты	Дата оплаты	-	Дата	"2023-05-25"
статус оплаты	Статус оплаты	-	Строка	"Оплачено"
период оплаты	Период, за который произведена оплата	-	Строка	"май 2023"

Таблица 8 – Атрибуты сущности «заселение\_выселение»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
id	Идентификатор операции заселения/выселения	>0	Число	1
id студента	Идентификатор студента	>0	Число	1
id комнаты	Идентификатор комнаты	>0	Число	101
тип операции	Тип операции (заселение или выселение)	-	Строка	"Заселение"
дата_операции	Дата операции	-	Дата	"2023-05-25"

причина	Причина выселения (если операция – выселение)	-	Строка	"Окончание обучения"
---------	---	---	--------	----------------------

Таблица 9 – Атрибуты сущности «мероприятия»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор мероприятия	>0	Число	1
название_мероприятия	Название мероприятия	-	Строка	"Новогодний концерт"
описание_мероприятия	Описание мероприятия	-	Строка	"Музыкальный концерт с участием студентов"
дата_и_время_проведения	Дата и время проведения мероприятия	-	Дата и время	"2023-12-31 20:00"
место_проведения	Место проведения мероприятия	-	Строка	"Концертный зал общешития"

Таблица 10 – Атрибуты сущности «запросы\_на\_вызов\_сотрудников»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор запроса	>0	Число	1
id_комнаты	Идентификатор комнаты, из которой поступил запрос	>0	Число	101
id_сотрудника	Идентификатор сотрудника, вызванного по запросу	>0	Число	3
тип_запроса	Тип запроса	-	Строка	"Ремонт"
дата_и_время_запроса	Дата и время запроса	-	Дата и время	"2023-05-25 10:00"

Таблица 11 – Атрибуты сущности «имущество\_комнат»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор предмета имущества	>0	Число	1
id_комнаты	Идентификатор комнаты, где находится предмет	>0	Число	101
название_предмета	Название предмета имущества	-	Строка	"Стол"
состояние_предмета	Состояние предмета	-	Строка	"Хорошее"

Таблица 12 – Атрибуты сущности «студенты\_электроустановки»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
<u>id</u>	Идентификатор записи	>0	Число	1

## Продолжение таблицы 12

id студента	Идентификатор студента	>0	Число	15
id_электроустановки	Идентификатор электроустановки	>0	Число	4

Первичные ключи, присвоенные созданным сущностям, выделены. Указанные атрибуты точно определяют соответствующие экземпляры сущности. Обнаруженные взаимосвязи и обоснование их приведены в таблице 12.

Таблица 13 – Связи между сущностями

Название первой сущности	Название второй сущности	Наименование связи	Тип связи	Описание типа связи
сотрудники	должности_сотрудников	Относятся к	Один ко многим	Каждой записи сущности «сотрудник» соответствует одна запись сущности «должность сотрудника».
студенты	комнаты	Проживают в	Один к одному	Каждой записи сущности «студент» соответствует одна запись сущности «комната».
электроустановки	студенты	Используются	Один ко многим	Каждой записи сущности «электроустановка» соответствует несколько записей сущности «студент».
студент_электроустановки	студенты, электроустановки	Соответствуют	Многие ко многим	Каждой записи сущности «студент_электроустановка» соответствуют записи сущностей «студент» и «электроустановка».
оплата	студенты	Оплачиваются	Один ко многим	Каждой записи сущности «оплата» соответствует несколько записей сущности «студент».
заселение_выселение	студенты, комнаты	Производятся	Многие ко многим	Каждой записи сущности «заселение_выселение» соответствуют записи сущностей «студент» и «комната».

запросы_на_вызов_сотрудников	комнаты, сотрудники	Формируются	Многие ко многим	Каждой записи сущности «запрос на вызов сотрудника» соответствуют записи сущностей «комната» и «сотрудник».
имущество комнат	комнаты	Находится в	Один ко многим	Каждой записи сущности «имущество комнаты» соответствует несколько записей сущности «комната».

Использование терминов "Один ко многим", "Многие ко многим" и "Один к одному" в контексте "Типа связи" относится к концепции моделирования баз данных, где "Один ко многим" означает, что одна запись в таблице может быть связана с несколькими записями в другой таблице, "Многие ко многим" означает, что множество записей в одной таблице может быть связано с множеством записей в другой таблице, а "Один к одному" означает, что одна запись в таблице связана с одной записью в другой таблице.

### 3.3.2 Логическое проектирование

На этапе логического проектирования создается реляционная логическая модель, в которой данные организуются в форме нормализованных отношений, учитывая связи между объектами предметной области и выполняя все необходимые преобразования для эффективной реализации в среде конкретной СУБД.

Реляционная логическая модель для автоматизированной системы управления общежитием для ФГБОУ ВО «АмГУ» была разработана с учетом всех выделенных сущностей: должности\_сотрудников, сотрудники, комнаты, студенты, электроустановки, студент\_электроустановки, оплата, заселение\_выселение, мероприятия, запросы\_на\_вызов\_сотрудников, имущество\_комнат.

Все сущности были преобразованы в наборы нормализованных отношений, каждое из которых представляет собой отдельную таблицу в базе данных. Отношения между этими сущностями были определены на основе связей в предметной области.

На рисунке 19 описана связь сущности «должности\_сотрудников» и сущности «сотрудники». Это отношение типа "один ко многим", где каждая должность может быть связана с несколькими сотрудниками. В данном случае, первичный ключ исходной сущности "должности\_сотрудников" добавляется в порожденную сущность "сотрудники" в качестве внешнего ключа.

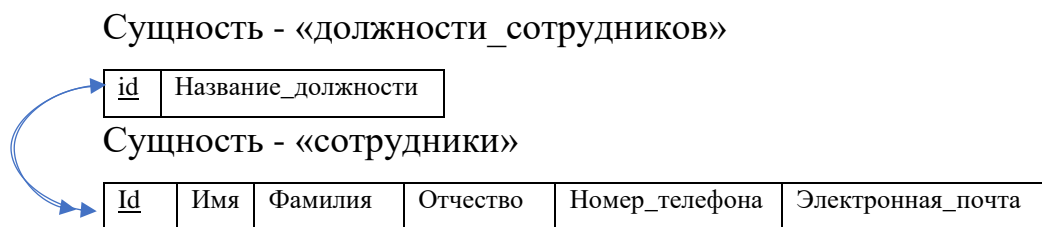


Рисунок 19 – Связь «должности\_сотрудников - сотрудники»

Отношение – «должности\_сотрудников»

<u>id</u>	Название_должности
-----------	--------------------

Отношение – «сотрудники»

<u>Id</u>	<u>Id_должности</u>	Имя	Фамилия	Отчество	Номер_телефона	Электронная_почта
-----------	---------------------	-----	---------	----------	----------------	-------------------

Рисунок 20 – Результат анализа связи «должности\_сотрудников - сотрудники»

На рисунке 21 описана связь сущности «комнаты» и сущности «студенты». Это отношение типа «один-к-одному». Сущность "студенты" проживает, в сущности, "комнаты" с типом связи "один к одному". Каждой записи сущности "студент" соответствует одна запись сущности "комната". В данном случае, первичный ключ "id\_комнаты" сущности "комнаты" добавляется в сущность "студенты" в качестве внешнего ключа.

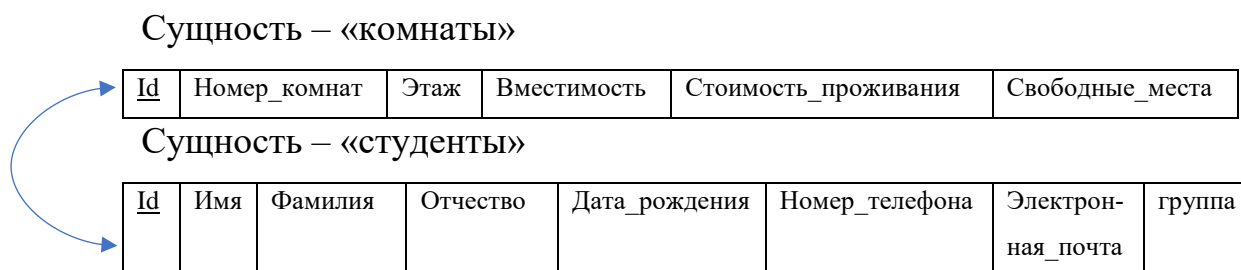


Рисунок 21 – Связь «комнаты - студенты»

Отношение – «комнаты»

<u>Id</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	--------------	------	-------------	----------------------	-----------------

Отношение – «студенты»

<u>Id</u>	<u>Id_комнаты</u>	Имя	Фамилия	Отчество	Дата_рождения	Номер_телефона	Электронная_почта	группа
-----------	-------------------	-----	---------	----------	---------------	----------------	-------------------	--------

Рисунок 22 – Результат анализа связи «комнаты - студенты»

На рисунке 23 описана связь сущности «электроустановки» и сущности «студенты». Сущность "электроустановки" используется сущностью "студенты" с типом связи "один ко многим". Каждой записи сущности "электроустановка" соответствует несколько записей сущности "студенты". В данном случае, первичный ключ "id\_студента" сущности "студенты" добавляется в сущность "электроустановки" в качестве внешнего ключа.

Сущность – «студенты»

<u>Id</u>	Имя	Фамилия	Отчество	Дата_рождения	Номер_телефона	Электронная_почта	Группа
-----------	-----	---------	----------	---------------	----------------	-------------------	--------

Сущность – «электроустановки»

<u>Id</u>	Тип	Стоимость_использования
-----------	-----	-------------------------



Рисунок 23 - Связь «электроустановки - студенты»

Отношение – «студенты»

<u>Id</u>	Имя	Фамилия	Отчество	Дата_рождения	Номер_телефона	Электронная_почта	Группа
-----------	-----	---------	----------	---------------	----------------	-------------------	--------

Отношение – «электроустановки»

<u>Id</u>	<u>Id студента</u>	Тип	Стоимость_использования
-----------	--------------------	-----	-------------------------

Рисунок 24 – Результат анализа связи «электроустановки - студенты»

На рисунке 25 описана связь сущности «электроустановки» и сущности «студенты». Сущность "студент\_электроустановки" представляет собой промежуточную сущность, которая служит для разрешения сложной двунаправленной



связи между сущностями "студенты" и "электроустановки". В данном случае, добавление промежуточной сущности позволяет установить связь между записями обеих сущностей через ключи, отражающие их соответствующие идентификаторы. Таким образом, для разрешения связи многие ко многим необходимо ввести промежуточную сущность "студент\_электроустановки" с ключами обеих сущностей "студенты" и "электроустановки". В данном случае, первичные ключи "id\_студента" и "id\_электроустановки" добавляются в сущность "студент\_электроустановки" в качестве внешних ключей.

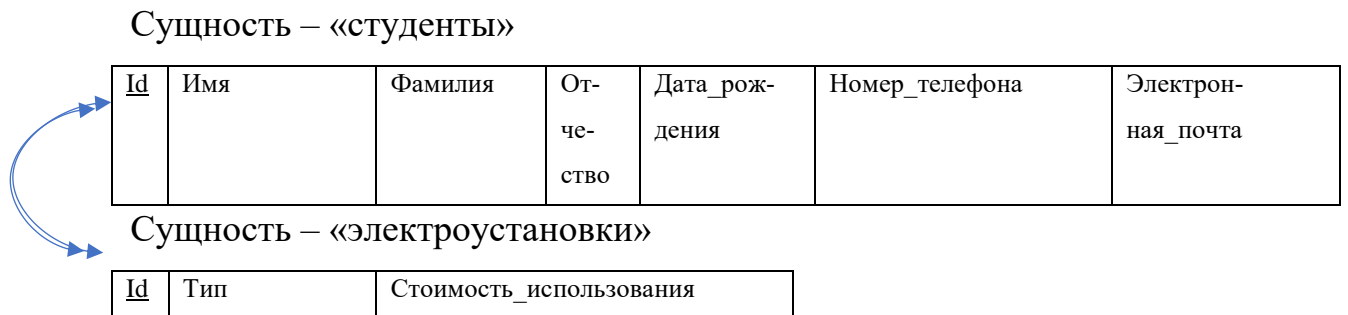


Рисунок 25 – Связь «электроустановки – студенты\_электроустановки - студенты»

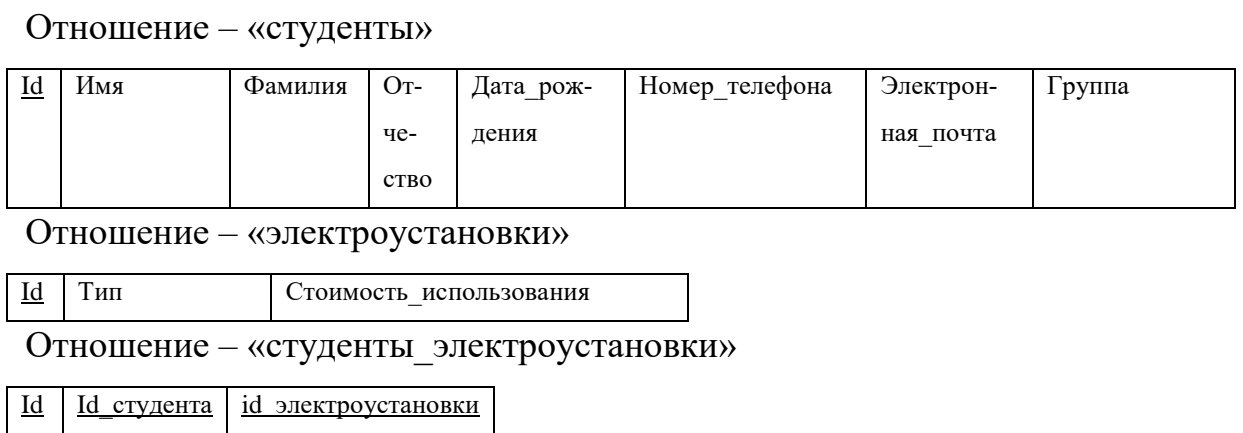


Рисунок 26 – Результат анализа связи «электроустановки – студенты\_электроустановки - студенты»

На рисунке 27 описана связь сущности «студенты» и сущности «оплата». Сущность "оплата" соответствует сущности "студенты" с типом связи "один ко многим". Каждой записи сущности "оплата" соответствует несколько записей сущности "студенты". В данном случае, первичный ключ "id\_студента" сущности "студенты" добавляется в сущность "оплата" в качестве внешнего ключа.

### Сущность – «студенты»

<u>Id</u>	Имя	Фамилия	От- че- ство	Дата_рож- дения	Номер_телефона	Электрон- ная_почта	Группа
-----------	-----	---------	--------------------	--------------------	----------------	------------------------	--------

### Сущность – «оплата»

<u>Id</u>	Сумма_оплаты	Дата_оплаты	Статус_оплаты	Период_оплаты
-----------	--------------	-------------	---------------	---------------

Рисунок 27 – Связь «студенты - оплата»

### Отношение – «студенты»

<u>Id</u>	Имя	Фамилия	От- че- ство	Дата_рож- дения	Номер_телефона	Электрон- ная_почта	Группа
-----------	-----	---------	--------------------	--------------------	----------------	------------------------	--------

### Отношение – «оплата»

<u>Id</u>	<u>Id_студента</u>	Сумма_оплаты	Дата_оплаты	Статус_оплаты	Период_оплаты
-----------	--------------------	--------------	-------------	---------------	---------------

Рисунок 28 – Результат анализа связи «студенты - оплата»

На рисунке 29 описана связь сущности «студенты» и сущности «комнаты». Для разрешения связи многие ко многим между сущностями "студенты" и "комнаты" необходимо ввести промежуточную сущность "заселение\_выселение". Эта промежуточная сущность служит для установления связи между записями обеих сущностей, позволяя отслеживать информацию о заселении и выселении студентов в комнаты. В данном случае, промежуточная сущность "заселение\_выселение" содержит внешние ключи, которые связывают идентификаторы студентов и комнат, участвующих в заселении и выселении. Таким образом, эта промежуточная сущность разрешает связь многие ко многим между сущностями "студенты" и "комнаты" и позволяет отслеживать и управлять информацией о процессе заселения и выселения студентов из комнат. Каждой записи сущности "заселение\_выселение" соответствуют записи сущностей "студент" и "комната". В данном случае, первичные ключи "id\_студента" и "id\_комнаты" добавляются в сущность "заселение\_выселение" в качестве внешних ключей.

### Сущность – «студенты»

<u>Id</u>	Имя	Фамилия	От- че- ство	Дата_рож- дения	Номер_телефона	Электрон- ная_почта	Группа
-----------	-----	---------	--------------------	--------------------	----------------	------------------------	--------

### Сущность – «комнаты»

<u>Id</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	--------------	------	-------------	----------------------	-----------------

Рисунок 29 – Связь «студенты - комнаты»

### Отношение – «студенты»

<u>Id</u>	Имя	Фамилия	От- че- ство	Дата_рож- дения	Номер_телефона	Электрон- ная_почта	Группа
-----------	-----	---------	--------------------	--------------------	----------------	------------------------	--------

### Отношение – «комнаты»

<u>Id</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	--------------	------	-------------	----------------------	-----------------

### Отношение – «комнаты»

<u>Id</u>	<u>Id_студента</u>	Id_комнаты	Тип_операции	Дата_операции	причина
-----------	--------------------	------------	--------------	---------------	---------

Рисунок 30 – Результат анализа связи «студенты – комнаты – заселение\_выселение»

На рисунке 31 описана связь сущности «комнаты» и сущности «сотрудники». Для разрешения связи многие ко многим между сущностями "комнаты" и "сотрудники" необходимо ввести промежуточную сущность "запросы\_на\_вызов\_сотрудников". Эта промежуточная сущность позволяет формировать связь между записями обеих сущностей, отражая информацию о запросах на вызов сотрудников для определенных комнат. В данном случае, промежуточная сущность "запросы\_на\_вызов\_сотрудников" содержит внешние ключи, которые связывают идентификаторы комнат и сотрудников, связанных с выполнением запросов. Таким образом, эта промежуточная сущность разрешает связь многие ко многим между сущностями "комнаты" и "сотрудники", позволяя отслеживать и управлять информацией о запросах на вызов сотрудников для определенных комнат. Сущность "запросы\_на\_вызов\_сотрудников" формируются сущностями "комнаты" и "сотрудники" с типом связи "многие ко многим". Каждой записи сущности "запрос на вызов сотрудника" соответствуют записи сущностей

"комната" и "сотрудник". В данном случае, первичные ключи "id\_комнаты" и "id\_сотрудника" добавляются в сущность "запросы\_на\_вызов\_сотрудников" в качестве внешних ключей.



Рисунок 31 – Связь «комнаты - сотрудники»

Отношение – «комнаты»

<u>Id</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	--------------	------	-------------	----------------------	-----------------

Отношение – «сотрудники»

<u>Id</u>	<u>Id_должности</u>	Имя	Фамилия	Отчество	Номер_телефона	Электронная_почта
-----------	---------------------	-----	---------	----------	----------------	-------------------

Отношение – «запросы\_на\_вызов\_сотрудников»

<u>Id</u>	<u>Id_комнаты</u>	<u>Id_сотрудника</u>	Описание_проблемы	Статус	Дата_создания_запроса	Дата_выполнения
-----------	-------------------	----------------------	-------------------	--------	-----------------------	-----------------

Рисунок 32 – Результат анализа связи «комнаты - сотрудники»

На рисунке 33 описана связь сущности «комнаты» и сущности «имущество\_комнат». Сущность "имущество\_комнат" находится в сущности "комнаты" с типом связи "один ко многим". Каждой записи сущности "имущество\_комнаты" соответствуют несколько записей сущности "комната". В данном случае, первичный ключ "id\_комнаты" сущности "комнаты" добавляется в сущность "имущество\_комнат" в качестве внешнего ключа.



Рисунок 33 – Связь «комнаты – имущество\_комнат»

Отношение – «комнаты»

<u>Id</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	--------------	------	-------------	----------------------	-----------------

Отношение – «имущество\_комнат»

<u>Id</u>	<u>Id_комнаты</u>	Номер_комнат	Этаж	Вместимость	Стоимость_проживания	Свободные_места
-----------	-------------------	--------------	------	-------------	----------------------	-----------------

Рисунок 34 – Результат анализа связи «комнаты – имущество\_комнат»

В результате логического проектирования и нормализации была получена логическая модель, которую можно посмотреть на рисунке 35.

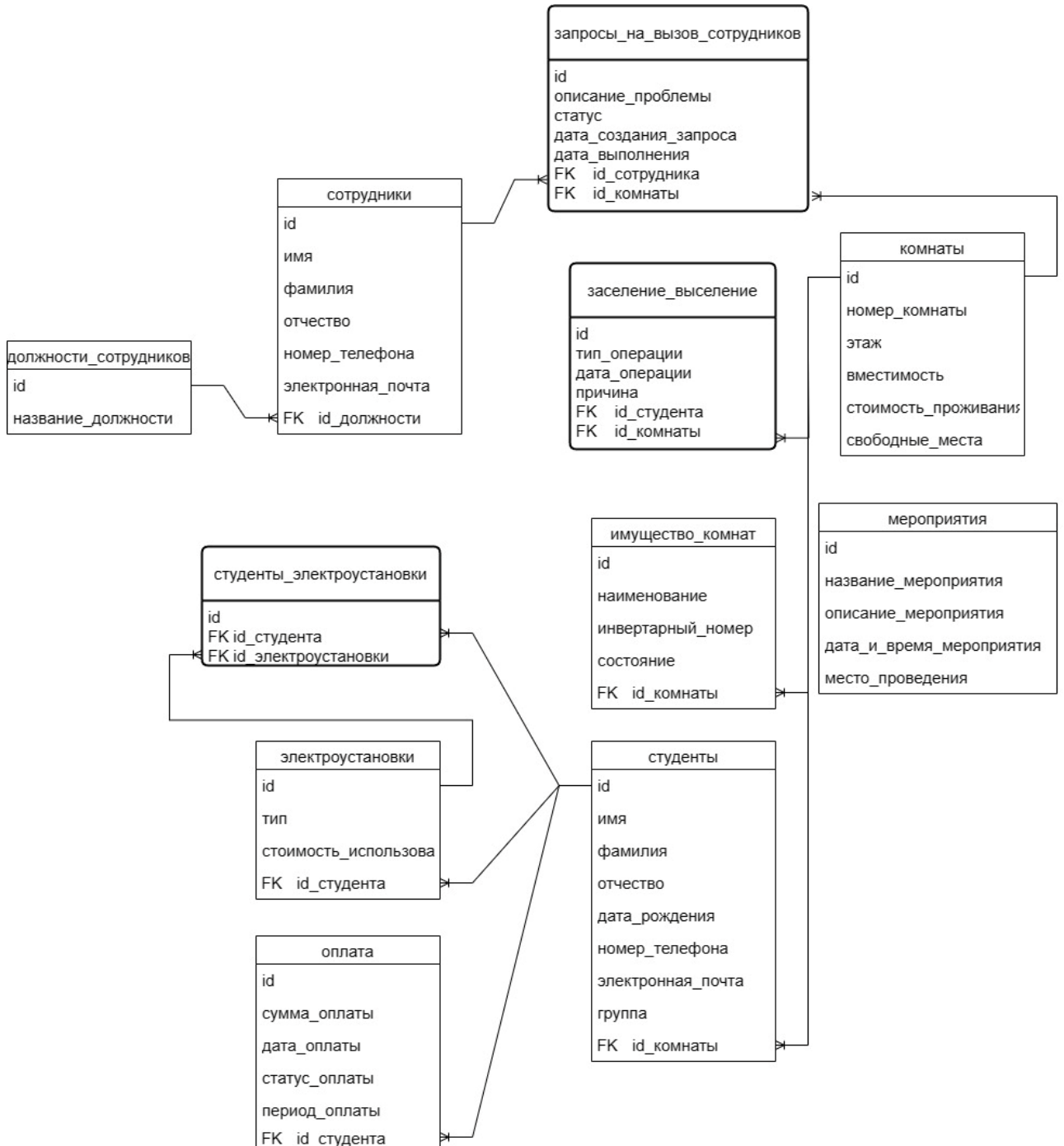


Рисунок 35 – Логическая модель базы данных для АСУ

### 3.3.3 Физическое проектирование

На этапе физического проектирования представляются проекты таблиц, которые будут реализованы в СУБД. На основании логической модели базы данных проектируется физическая модель. Первая из них расположена в таблице 14.

Таблица 14 – Спецификации таблицы «должности\_сотрудников»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да (совпадения не допускаются)	нет
название_должности	varchar(255)	-	нет	нет

Таблица 15 – Спецификации атрибутов сущности «сотрудники»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да (совпадения не допускаются)	нет
имя	varchar(255)	-	нет	нет
фамилия	varchar(255)	-	нет	нет
отчество	varchar(255)	-	нет	да
номер телефона	varchar(255)	-	нет	да
электронная почта	varchar(255)	-	нет	да
id должности	integer	>0 и соответствие id в таблице "должности сотрудников"	да	да

Таблица 16 – Спецификации атрибутов сущности «комнаты»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да (совпадения не допускаются)	нет
номер комнаты	varchar(10)	-	нет	нет
этаж	integer	>0	нет	нет
вместимость	integer	>0	нет	нет
стоимость_проживания	decimal(10,2)	>0	нет	нет
свободные места	integer	>=0 и <= вместимость	нет	нет

Таблица 17 – Спецификации атрибутов сущности «студенты»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
имя	VARCHAR(255)	-	Нет	Нет
фамилия	VARCHAR(255)	-	Нет	Нет
отчество	VARCHAR(255)	-	Нет	Да
дата рождения	DATE	-	Нет	Нет
номер телефона	VARCHAR(255)	-	Нет	Да
электронная почта	VARCHAR(255)	-	Нет	Да
группа	VARCHAR(255)	-	Нет	Нет

id комнаты	INTEGER	>0	Нет	Да
------------	---------	----	-----	----

Таблица 18 – Спецификации атрибутов сущности «электроустановки»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
тип	VARCHAR(255)	-	Нет	Нет
стоимость_использования	DECIMAL(10,2)	>=0	Нет	Нет
id студента	INTEGER	>0	Нет	Да

Таблица 19 – Спецификации атрибутов сущности «студент\_электроустановки»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
id студента	INTEGER	>0	Нет	Нет
id электроустановки	INTEGER	>0	Нет	Нет

Таблица 20 – Спецификации атрибутов сущности «оплата»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
id студента	INTEGER	>0	Нет	Нет
сумма оплаты	DECIMAL(10,2)	>=0	Нет	Нет
дата оплаты	DATE	-	Нет	Нет
статус оплаты	VARCHAR(255)	-	Нет	Нет
период оплаты	VARCHAR(255)	-	Нет	Нет

Таблица 21 – Спецификации атрибутов сущности «заселение\_выселение»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
id студента	INTEGER	>0	Нет	Нет
id комнаты	INTEGER	>0	Нет	Нет
тип операции	VARCHAR(255)	-	Нет	Нет
дата операции	DATE	-	Нет	Нет
причина	VARCHAR(255)	-	Нет	Да

Таблица 22 – Спецификации атрибутов сущности «запросы\_на\_вызов\_сотрудников»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
-------------------	------------	------------------------------------	------------	-------------------

Продолжение таблицы 22

id	SERIAL	>0	Да (совпадения не допускаются)	Нет
id комнаты	INTEGER	>0	Нет	Нет
id сотрудника	INTEGER	>0	Нет	Нет
описание_проблемы	TEXT	-	Нет	Нет
статус	VARCHAR(255)	-	Нет	Нет
дата_создания_запроса	DATE	-	Нет	Нет
дата_выполнения	DATE	-	Нет	Да

Таблица 23 – Спецификации атрибутов сущности «мероприятия»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
название_мероприятия	VARCHAR(255)	-	Нет	Нет
описание_мероприятия	TEXT	-	Нет	Да
дата и время проведения	TIMESTAMP	-	Нет	Нет
место проведения	VARCHAR(255)	-	Нет	Нет

Таблица 24 – Спецификации атрибутов сущности «имущество\_комнат»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	SERIAL	>0	Да (совпадения не допускаются)	Нет
наименование	VARCHAR(255)	-	Нет	Нет
инвентарный_номер	VARCHAR(255)	-	Нет	Нет
состояние	VARCHAR(255)	-	Нет	Нет
id комнаты	INTEGER	>0	Нет	Нет

В завершение процесса физического проектирования были созданы таблицы, которые будут использоваться для внедрения базы данных. Описанные таблицы отображают все необходимые атрибуты, определенные в ходе логического проектирования, с учетом специфических требований, предъявляемых к реляционным СУБД, таких как PostgreSQL.

Таблицы были спроектированы с учетом следующих ключевых аспектов:

- использование подходящих типов данных для каждого атрибута;
- установление ограничений на допустимые значения, включая ограничения на NULL и уникальность, где это применимо;



- использование индексации для ключевых атрибутов, что облегчает поиск и обеспечивает эффективное использование ресурсов базы данных;

- определение внешних ключей для обеспечения целостности данных и связей между таблицами;

Окончательная физическая модель базы данных представлена на рисунке 36. Эта модель отображает структуру каждой таблицы, включая все атрибуты и связи. На данном этапе проекта базы данных выполнена подготовка всех необходимых материалов для реализации базы данных в выбранной СУБД - PostgreSQL.

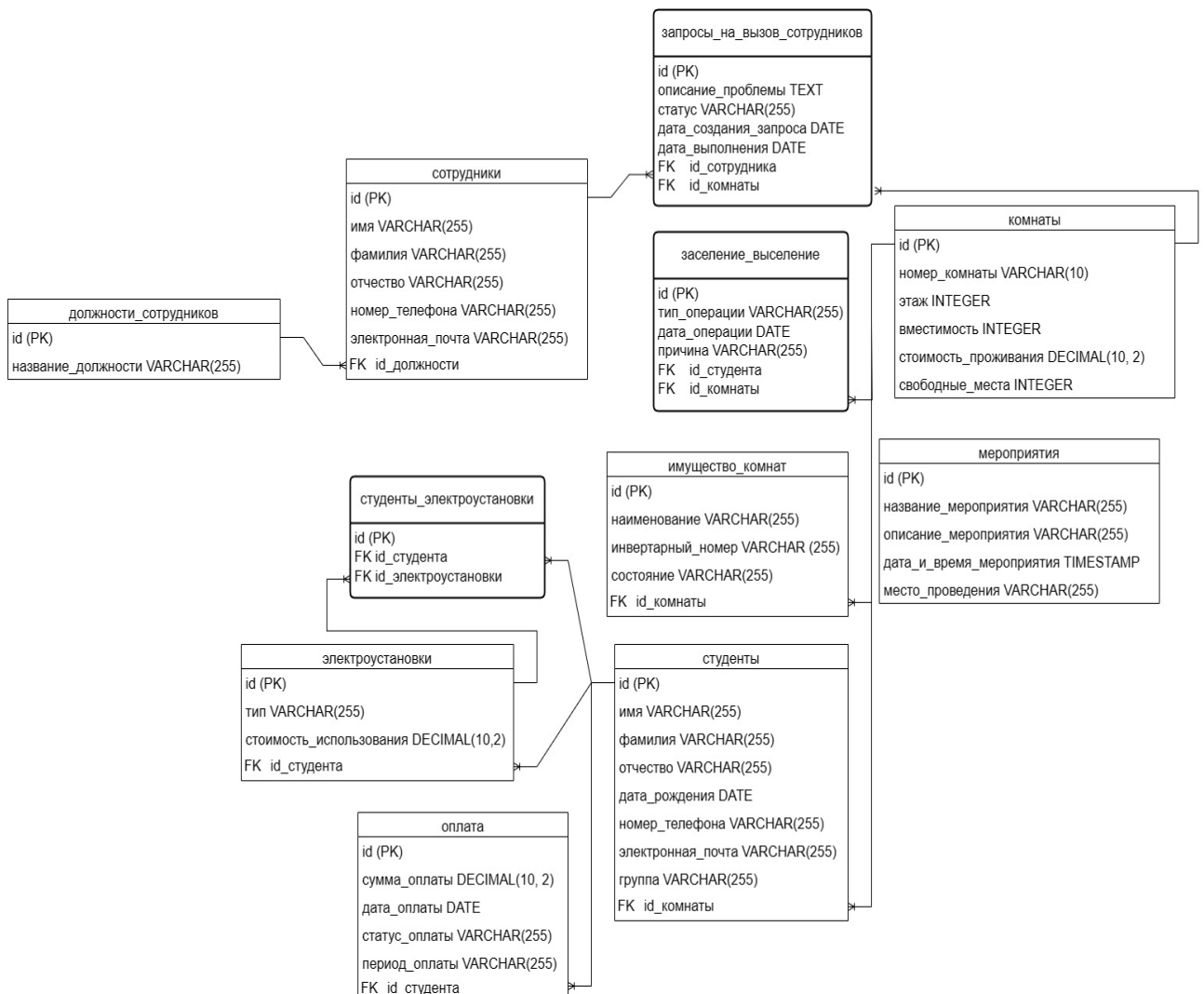


Рисунок 36 – Физическая модель базы данных АСУ

### 3.4 Проектирование базы данных для пользователей АСУ

#### 3.4.1 Инфологическое проектирование

Кроме основной базы данных, описывающей функционирование общежития ФГБОУ ВО «АмГУ», было решено создать дополнительную базу данных, предназначенную для управления доступом пользователей к информационным ресурсам автоматизированной системы управления. Это было обусловлено необходимостью разграничивать права доступа сотрудников общежития и обеспечивать безопасность хранения и обработки информации.

Для этого были выделены три основные сущности: пользователи, права\_доступа и роли.

Таблица 25 – Сущности базы данных пользователей

Название	Описание
пользователи	Содержит информацию о зарегистрированных пользователях системы, включая их имя, зашифрованный пароль и роль
права_доступа	Содержит информацию о правах доступа пользователей к различным базам данных системы
роли	Содержит информацию о ролях пользователей в системе

Каждая из этих сущностей содержит ряд атрибутов, которые предназначены для хранения различных видов информации. При проектировании этих таблиц были учтены необходимые требования к целостности данных и соблюдены принципы нормализации для обеспечения эффективности и надежности базы данных.

Таблица 26 – Атрибуты сущности «пользователи»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
id	Идентификатор пользователя	>0	Число	1
имя_пользователя	Имя пользователя	-	Строка	"Ivanov"
хэш пароля	Хэш пароля пользователя	-	Строка	"5f4dcc3b5aa765d61d8327deb882cf99"
соль	Соль для хэша пароля	-	Строка	"random_string"
id роли	Идентификатор роли пользователя	>0	Число	2

Таблица 27 – Атрибуты сущности «права\_доступа»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
-------------------	-------------------	-------------------	-------------------	-----------------

id	Идентификатор записи	>0	Число	1
id пользователя	Идентификатор пользователя	>0	Число	3
название_бд	Название базы данных	-	Строка	"users_db"

Таблица 28 – Атрибуты сущности «роли»

Название атрибута	Значение атрибута	Диапазон значений	Единицы измерения	Пример атрибута
id	Идентификатор роли	>0	Число	1
название роли	Название роли	-	Строка	"Зав.общезижем"

Первичные ключи, присвоенные созданным сущностям, выделены. Указанные атрибуты точно определяют соответствующие экземпляры сущности. Обнаруженные взаимосвязи и обоснование их приведены в таблице 29.

Таблица 29 – Связи между сущностями

Название первой сущности	Название второй сущности	Наименование связи	Тип связи	Описание типа связи
пользователи	роли	Принадлежат к	Один к одному	Каждой записи сущности «пользователь» соответствует одна запись сущности «роль».
права_доступа	пользователи	Принадлежат к	Один к одному	Каждой записи сущности «права_доступа» соответствует одна запись сущности «пользователь».
пользователи	права_доступа	Имеют	Один ко многим	Каждой записи сущности «пользователь» соответствует несколько записей сущности «права_доступа».

### 3.4.2 Логическое проектирование

Реляционная логическая модель для автоматизированной системы управления общежитием ФГБОУ ВО «АмГУ» была разработана с учетом всех выделенных сущностей: пользователи, права\_доступа, роли.

Все сущности были преобразованы в наборы нормализованных отношений, каждое из которых представляет собой отдельную таблицу в базе данных.

Отношения между этими сущностями были определены на основе связей в предметной области.

На рисунке 37 описана связь сущности «пользователи» и сущности «роли». Это отношение типа "один к одному", где каждый пользователь может быть связан только с одной ролью. В данном случае, первичный ключ исходной сущности "роли" добавляется в порожденную сущность "пользователи" в качестве внешнего ключа.



Рисунок 37 – Связь «роли - пользователи»

Отношение – «роли»

<u>Id</u>	Название_роли
-----------	---------------

Отношение – «пользователи»

<u>Id</u>	<u>Id_роли</u>	Имя_пользователя	Хэш_пароля	Соль
-----------	----------------	------------------	------------	------

Рисунок 38 – Результат анализа связи «роли - пользователи»

На рисунке 39 представлена связь между сущностями "пользователи" и "права\_доступа". Это отношение "один ко многим", где каждому пользователю могут быть назначены различные уровни доступа к разным базам данных. Первичный ключ исходной сущности "пользователи" добавляется в порожденную сущность "права\_доступа" в качестве внешнего ключа.

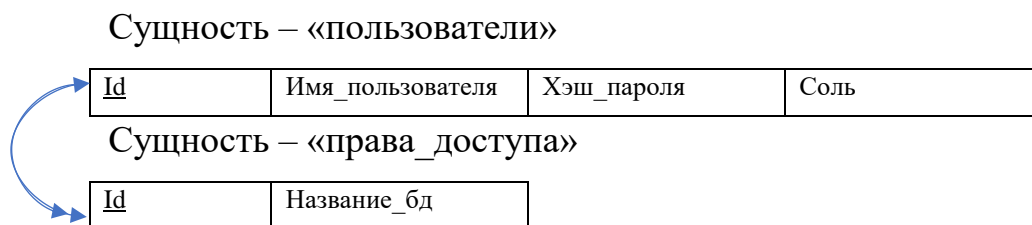


Рисунок 39 – Связь «пользователи – права\_доступа»

Отношение – «пользователи»

<u>Id</u>	Имя_пользователя	Хэш_пароля	Соль
-----------	------------------	------------	------

Отношение – «права\_доступа»

<u>Id</u>	<u>Id_пользователя</u>	Название_бд
-----------	------------------------	-------------

Рисунок 40 – Результат анализа связи – «пользователи – права\_доступа»

В результате логического проектирования и нормализации была получена логическая модель, которую можно посмотреть на рисунке 41. Для построения логической модели базы данных использовались базовый инструменты PostgreSQL – ERDTools.

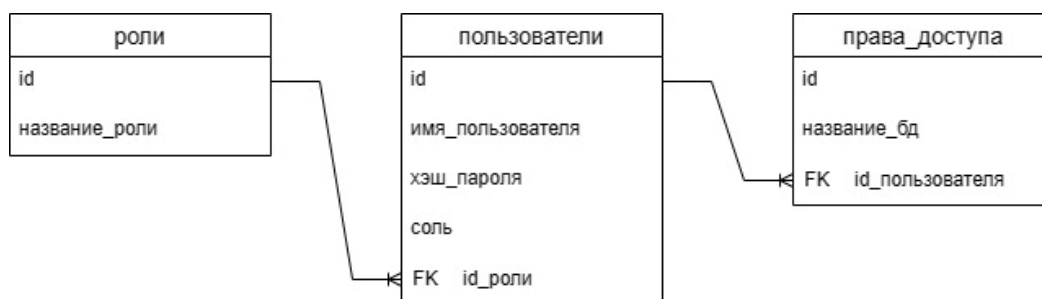


Рисунок 41 – Логическая модель базы данных

### 3.4.3 Физическое проектирование

На этапе физического проектирования представляются проекты таблиц, которые будут реализованы в СУБД. На основании логической модели базы данных проектируется физическая модель. Спецификации таблиц представлены в таблицах 30-32.

Таблица 30 – Спецификации атрибутов сущности «пользователи»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да	нет
имя_пользователя	varchar(255)	-	да	нет
хэш_пароля	varchar(255)	-	нет	нет
соль	varchar(255)	-	нет	нет
id_роли	integer	-	нет	да

Таблица 31 – Спецификации атрибутов сущности «права\_доступа»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да	нет
имя_пользователя	varchar(255)	-	да	нет
хэш_пароля	varchar(255)	-	нет	нет
соль	varchar(255)	-	нет	нет
id_роли	integer	-	нет	да

id	integer	>0	да	нет
id пользователя	integer	>0	нет	нет
название бд	text	-	нет	нет

Таблица 32 – Спецификации атрибутов сущности «роли»

Название атрибута	Тип данных	Ограничения на допустимые значения	Индексация	Допустимость NULL
id	integer	>0	да	нет
название роли	varchar(255)	-	да	нет

Окончательная физическая модель базы данных представлена на рисунке 42. Эта модель отображает структуру каждой таблицы, включая все атрибуты и связи. На данном этапе проекта базы данных выполнена подготовка всех необходимых материалов для реализации базы данных в выбранной СУБД - PostgreSQL.

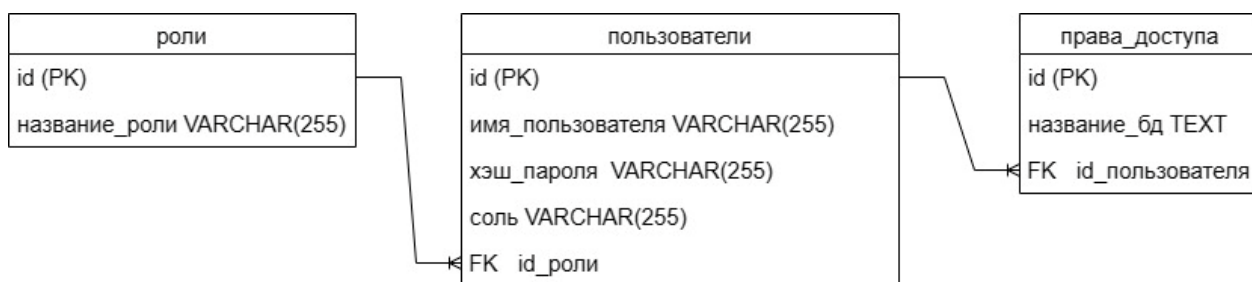


Рисунок 42 – Физическая модель базы данных

## 4 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ОБЩЕЖИТИЕМ

### 4.1 Описание модулей программы

Описание модулей программы АСУ представлено в таблице 33.

Таблица 33 – Описание модулей программы

Системное наименование	Наименование	Функциональное назначение	Входные данные	Выходные данные
LoginWindow.xaml.cs	Окно входа	Авторизация пользователя	Таблица пользователи, роли, права доступа	Таблица пользователи
AuthAdminWindow.xaml.cs	Окно аутентификации администратора	Авторизация администратора	Таблица пользователи, роли, права доступа	Таблица пользователи, права доступа
AdminWindow.xaml.cs	Окно администратора	Навигация по программе, управление правами пользователей, выдачи новых прав, и добавление новых пользователей.	Таблицы: пользователи, роли, права доступа	Те же таблицы, но с обновленными данными
EditUserAccessWindow.xaml.cs	Окно изменения доступа пользователя	Управление правами доступа пользователя	Таблица пользователи, права доступа, роли	Таблица пользователи с обновленными правами доступа
MainWindow.xaml.cs	Главное окно	Общий обзор и навигация по функциям	Таблица пользователи, комнаты, студенты, мероприятия	-
ResidentsWindow.xaml.cs	Окно проживающих	Просмотр и управление данными о проживающих, оплатой, заселением, выселением, переселением, электроустановок	Таблица студенты, комнаты, заселение выселение, электроустановки, оплата	Таблица студенты с обновленными данными
StudentEditWindow.xaml.cs	Окно редактирования студентов	Редактирование информации о студентах	Таблица студенты	Таблица студенты с обновленной информацией
ElectricalAppliancesStudents.xaml.cs	Окно электроустановок студентов	Управление электроустановками студентов	Таблицы: студенты, электроустановки	Те же таблицы, но с обновленными данными

Продолжение таблицы 33

SendProgressWindow.xaml.cs	Окно отправки прогресса	Отправка отчетов о прогрессе	Таблица запросы_на_вызов_сотрудников, оплата	Окно с обновленными данными о прогрессе отправки сообщений
RoomsWindow.xaml.cs	Окно комнат	Просмотр и управление комнатами	Таблица комнаты, имущество_комнат, студенты	Таблица комнаты с обновленными данными
TenantsRoom.xaml.cs	Окно проживающих в комнаты	Управление данными о проживании	Таблицы: студенты, комнаты	Те же таблицы, но с обновленными данными
PropertyRoom.xaml.cs	Окно имущества комнаты	Управление имуществом комнаты	Таблица имущество_комнат, комнаты	Таблица имущество_комнат с обновленными данными
RequestRepair.xaml.cs	Окно запроса на ремонт	Создание запроса на ремонт	Таблицы: комнаты, запросы_на_вызов_сотрудников	Таблица запросы_на_вызов_сотрудников с новым запросом
ListEmployee.xaml.cs	Окно списка сотрудников	Просмотр списка сотрудников	Таблица сотрудники	-
AddEmployee.xaml.cs	Окно добавления сотрудника	Добавление нового сотрудника	Таблица сотрудники, должности сотрудников	Таблица сотрудники с новым сотрудником
AddProperty.xaml.cs	Окно добавления имущества	Добавление нового имущества	Таблица имущество_комнат	Таблица имущество_комнат с новым имуществом
EventsWindow.xaml.cs	Окно мероприятий	Управление данными о планируемых мероприятиях	Таблица мероприятия	Таблица мероприятия с обновленными данными

На рисунке 43 изображена упрощенная структура программы, которая описывает основные модули программы и базы данных, с которыми она взаимодействует.



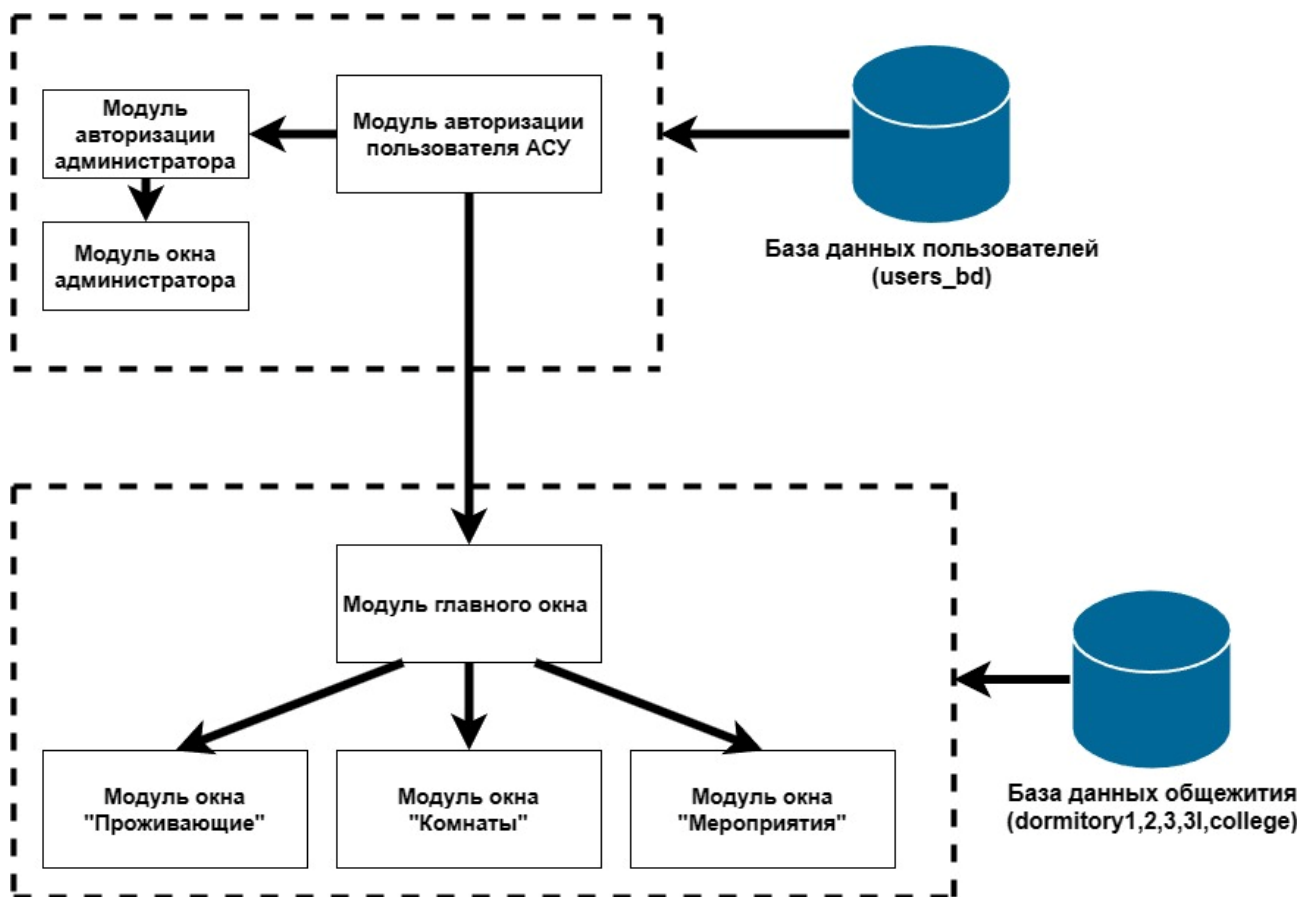


Рисунок 43 – Упрощенная структура программы

#### 4.2 Алгоритм функционирования программы

Автоматизированная система управления общежитием включает в себя два основных режима работы - режим администратора и режим обычного пользователя:

- режим администратора: для входа в этот режим с формы авторизации нажимается кнопка "Панель администратора". Затем происходит ввод логина и пароля. Если они верны, открывается панель администратора, где имеется возможность управления пользователями и настройка их прав доступа к базам данных общежитий. В случае ввода неверных данных, система возвращает пользователя обратно на форму авторизации;
- режим пользователя: если кнопка панели администратора не нажата, происходит ввод логина и пароля пользователя. При успешной авторизации открывается главная форма, где пользователь может просматривать информацию,

доступную ему согласно установленным правам доступа. В случае ошибки в логине или пароле также происходит возврат к форме авторизации;

На рисунке 44 изображена упрощенная схема алгоритма работы программы. Упрощенной эта схема является по той причине, что функционал программы имеет довольно много различных функций, которые не отобразить на одной схеме. Однако, алгоритм, приведенный ниже, отображает в общих чертах алгоритм работы программы.

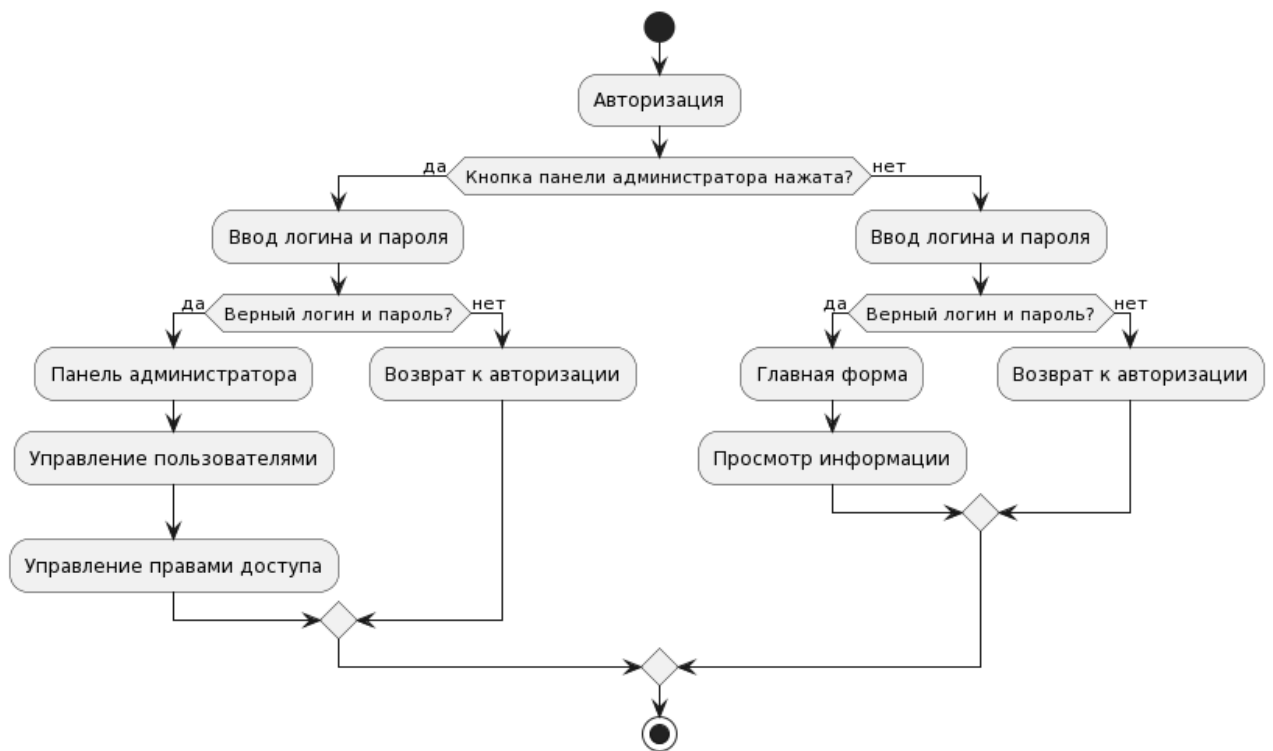


Рисунок 44 – Упрощенная схема алгоритма работы программы

### 4.3 Описание пользовательского интерфейса программы

#### 4.3.1 Модуль авторизации и окна администратора

Модуль авторизации представляет собой интерфейс, который позволяет пользователю и администратору войти в систему. Этот модуль изображен на рисунке Б.1 в приложении Б.

Этот модуль состоит из формы авторизации, которая включает в себя два поля ввода - для логина и пароля. Помимо полей ввода на форме расположены две кнопки - "Вход" и "Панель администратора". При нажатии на кнопку "Вход"

происходит проверка введенных данных и, в случае успеха, переход в основное окно программы под учетной записью пользователя. Если введенные данные некорректны, система уведомляет об этом пользователя (см. рисунок Б.2 в приложении Б).

Кнопка "Панель администратора" предназначена для перехода в режим администрирования системы. При ее нажатии также происходит проверка введенных данных в окне авторизации (см. рисунок Б.3 в приложении Б) и, если логин и пароль относятся к учетной записи администратора, открывается окно администратора (см. рисунок Б.4 в приложении Б).

Окно администратора содержит интерфейс для управления пользователями системы. Администратор может просматривать список пользователей, изменять их права доступа (см. рисунок Б.4 в приложении Б), добавлять новых пользователей и удалять существующих. Все изменения сохраняются в системе, и они вступают в силу сразу после их применения.

Интерфейс окна администратора представляет собой таблицу со списком пользователей и кнопки для добавления нового пользователя и удаления, выбранного из списка. Каждый пользователь в таблице имеет информацию о своем логине, роли и текущих правах доступа.

Этот модуль обеспечивает безопасность и управление доступом к функционалу системы.

#### 4.3.2 Главная форма программы

На рисунке 45 изображена главная форма программы, которая представляет собой центральное место для управления всеми функциями приложения, включая информацию о студентах, комнатах и мероприятиях. В этом интерфейсе представлены следующие основные разделы:

- панель состояния: этот раздел содержит общую информацию о статусе общежития, включая количество проживающих студентов, количество свободных комнат и текущие задолженности. Эта информация связана с соответствующими свойствами и автоматически обновляется при изменении этих свойств;

– календарь и список задач: здесь представлен календарь и список ближайших мероприятий, которые помогают в планировании и организации деятельности общежития;

– разделы и вкладки: в этом разделе представлены три основных вкладки: "Проживающие", "Комнаты" и "Мероприятия". Каждая вкладка содержит специализированный интерфейс для управления своими соответствующими элементами;

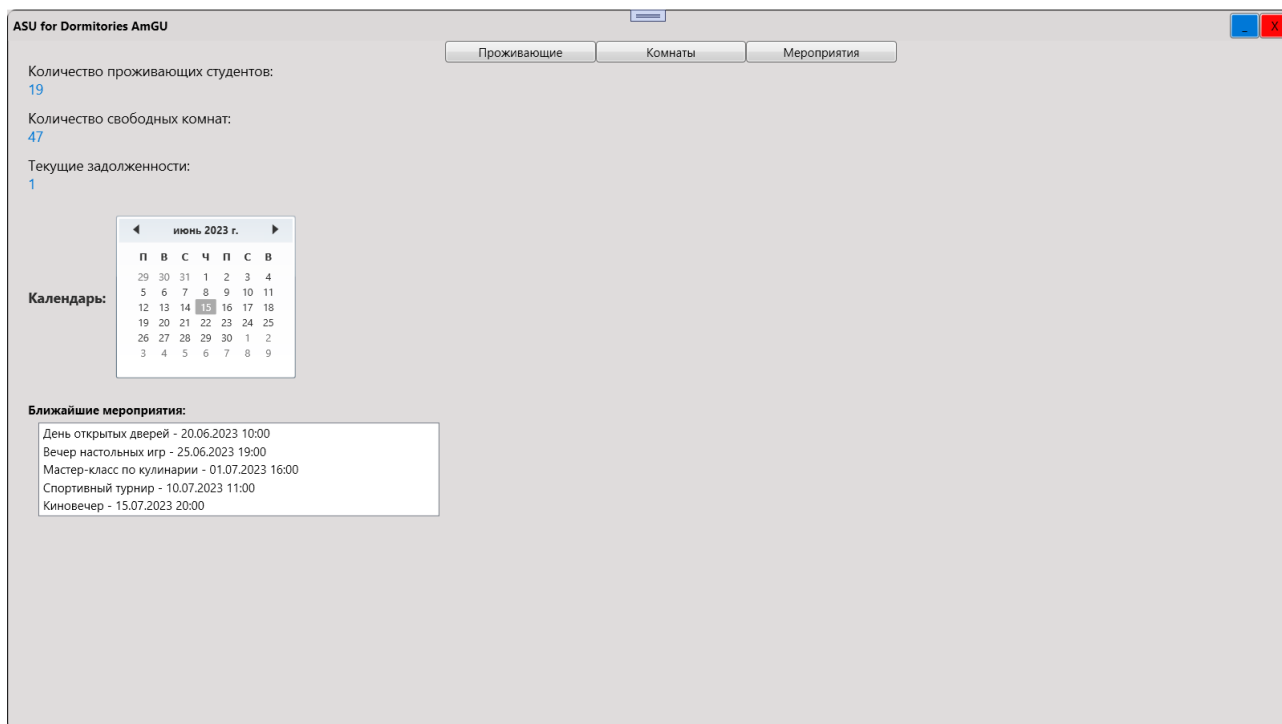


Рисунок 45 – Главная форма программы

#### 4.3.3 Модуль «Проживающие»

Модуль "Проживающие" представляет собой интерфейс, который обеспечивает управление данными о проживающих в общежитии студентах.

Этот модуль содержит разделы: "Просмотр и редактирование данных студентов", "Управление оплатой", "Заселение, выселение и переселение", и "Отчет по заселениям, выселениям и переселениям".

В разделе "Просмотр и редактирование данных студентов" (см. рисунок В.1 в приложении В) отображается таблица, которая включает в себя основную информацию о студентах, такую как ФИО, дата рождения, номер телефона, электронная почта, группа и номер комнаты. Также присутствуют инструменты для

поиска студентов по группе и номеру комнаты, а также кнопки для добавления, редактирования и удаления записей о студентах.

Раздел "Управление оплатой" (см. рисунок В.2 в приложении В) предназначен для управления информацией о платежах студентов за проживание в общежитии.

Здесь можно составить отчет по оплате и задолженностям и сохранить в формате документа Excel, применить фильтры для поиска платежей и должников, а также разослать квитанции всем должникам (см. рисунок В.3 в приложении В). Задолженности начисляются каждое 10-е число нового месяца в автоматическом режиме.

"Заселение, выселение и переселение" (см. рисунок В.7 в приложении В) представляет собой функционал для управления процессами заселения, выселения и переселения студентов. Здесь можно выбрать студента из таблицы и выполнить соответствующую операцию. Также есть возможность открыть список электроустановок заселенного студента для добавления используемых им электроустановок для суммирования стоимости их использования с оплатой за проживание в общежитии (см. рисунок В.8 в приложении В).

В разделе "Отчет по заселениям, выселениям и переселениям" (см. рисунок В.6 в приложении В) можно составить отчет по выполненным операциям заселения, выселения и переселения за выбранный период времени и сохранить его в формате документа Excel в удобной папке.

Каждый из этих разделов обеспечивает гибкое управление информацией о проживающих в общежитии студентах и выполняемых операциях.

#### 4.3.4 Модуль «Комнаты»

Модуль "Комнаты" представляет собой интерфейс, который обеспечивает управление данными о комнатах общежития. Этот модуль содержит два основных раздела: "Управление комнатами" и "Вызов сотрудников".

В разделе "Управление комнатами" (см. рисунок Г.1 в приложении Г) отображается таблица, содержащая информацию о комнатах, включая номер комнаты, этаж, вместимость, стоимость проживания и количество свободных мест.

Кроме того, каждая строка таблицы включает две кнопки, которые позволяют просмотреть имущество комнаты (см. рисунок Г.2 в приложении Г) и список текущих жильцов (см. рисунок Г.3 в приложении Г). В этом разделе также предоставлены функции для создания отчетов о доступности комнат.

Раздел "Вызов сотрудников" (см. рисунок Г.4 в приложении Г) обеспечивает управление запросами на вызов сотрудников для решения различных вопросов, связанных с комнатами. Здесь можно создать новый запрос (см. рисунок Г.5 в приложении Г), очистить журнал запросов и открыть список всех сотрудников общежития. Таблица запросов содержит информацию о номере комнаты, должности сотрудника, его полном имени, описании проблемы, дате создания запроса и его статусе. Пользователь может изменить статус запроса с помощью соответствующего чекбокса в таблице.

#### 4.3.5 Модуль «Мероприятия»

Модуль "Мероприятия" представляет собой интерфейс для организации плана мероприятий в общежитии. Этот модуль включает в себя три основных раздела: "Планирование мероприятий", "Оповещения о мероприятиях" и "Отчет по мероприятиям".

В разделе "Планирование мероприятий" пользователь может добавить новое мероприятие, указав его название, описание, дату и время проведения, а также место проведения (см. рисунок Д.1 в приложении Д). Данный раздел предназначен для упрощения процесса организации мероприятий и обеспечивает хранение всей необходимой информации о планируемых мероприятиях в одном месте.

Раздел "Оповещения о мероприятиях" предоставляет возможность установить настройки оповещений о мероприятиях. Здесь пользователь может выбрать тип оповещения (например, отправить на почту) и ввести адрес электронной почты, на который будут приходить оповещения (см. рисунок Д.2 в приложении Д). В разделе "Отчет по мероприятиям" пользователь может создать отчет о проведенных мероприятиях за выбранный период времени.

## 5 БЕЗОПАСНОСТЬ

### 5.1 Информационная безопасность

В современном мире, когда данные становятся основой для большинства процессов и операций, важность информационной безопасности не может быть недооценена. В области информационной безопасности, основными принципами являются: целостность, доступность и конфиденциальность.

Целостность данных обеспечивает, что информация не может быть изменена незаметно, и только авторизованные пользователи могут вносить изменения. В системе АСУ для общежитий АмГУ целостность данных обеспечивается через использование разграничения доступа на основе ролей. Каждый пользователь имеет определенные привилегии, которые определяют, какие данные они могут просматривать или изменять.

Доступность гарантирует, что информация доступна для использования, когда это необходимо. Система обеспечивает доступность данных через резервное копирование, что позволяет восстановить данные в случае сбоя или иной потери информации. Кроме того, система проектировалась с учетом производительности и масштабируемости, чтобы обеспечить быстрый доступ к информации, даже при большом объеме данных.

Конфиденциальность обеспечивает, что информация доступна только тем, кому она действительно нужна. В нашей системе конфиденциальность поддерживается через использование авторизации и аутентификации, шифрования данных при передаче и параметризацию SQL запросов, а также хеширования паролей.

Вместе эти три принципа обеспечивают высокий уровень информационной безопасности, защищая данные от несанкционированного доступа, изменений, утраты или утечки. В этом разделе мы подробно рассмотрим, как эти принципы реализуются в рамках разрабатываемой системы.

#### 5.1.1 Разграничение доступа

В системе реализована модель доступа на основе ролей, которая определяет уровень доступа каждого пользователя к различным базам данных, как указано в таблице 34. Например, администратор имеет доступ ко всем базам данных, в то время как заведующий имеет доступ только к базам данных конкретного общежития. Это обеспечивает необходимую гибкость при сохранении общего принципа наименьших привилегий.

Таблица 34 – Права доступа в программе

Вид пользователя	Роль пользователя	Права доступа к базе данных
Администратор АСУ	Администратор	dormitory1, dormitory2, dormitory3, dormitory3L, college
Заведующий	Зав.общежитием №1	dormitory1
Заведующий	Зав.общежитием №2	dormitory2
Заведующий	Зав.общежитием №3	dormitory3
Заведующий	Зав.общежитием №3Л	dormitory3L
Заведующий	Зав.общежитием колледжа	college

### 5.1.2 Аутентификация, Авторизация и Шифрование

*Аутентификация и авторизация:* в системе реализована система аутентификации, которая обеспечивает доступ к данным и функциональности только для авторизованных пользователей. Это включает в себя использование уникальных идентификаторов и паролей для входа в систему.

*Хранение и обработка паролей:* при создании нового пользователя, вводимый пароль проходит через процесс хеширования, а затем хэш-код, вместе с солью, сохраняется в базе данных. При аутентификации пользователя, введенный пароль повторно проходит через тот же процесс хеширования с использованием сохраненной соли, и полученный хэш сравнивается с сохраненным в базе данных. Пароли пользователей никогда не хранятся в открытом виде, что повышает уровень защиты данных. На рисунке 46 изображена таблица пользователей в базе данных пользователей АСУ, где можно видеть хэш пароля и его «соль».



id [PK] integer	имя_пользователя character varying (255)	хэш_пароля character varying (255)	соль character varying (255)	id_поли integer
17	admin	G8itbJvMqLmXdQ57vQmwXSuPjh...	8Vp0TEJ4hHvFyuCh5u3h7Q==	6
18	Zav1	k58cMvEo+lnM1wlrBSWBPOqcGv...	K7gBSccdj/VHf+Qy5lSc0A==	1
19	Zav2	loBHP1ST00EErGKwFjjK6zmW09...	zn8M7FZ1QRPbAuodciA1FQ==	2
20	Zav3	Vfwp9zdxm4QDKVyXBsBTpbRjEh...	5mTAuaaA4CZc6V1Ajs3ftA==	3
21	Zav3L	kmgwQ/8aKclBac7/n85E/yDj87Y=	+kYm9mwJsnDghZBSyzEt/g==	4
22	ZavCollege	LABWcYlMSEUuruHrXaDx8XCo8...	/+XB1SLsqjtjwLh5NWXzC9g==	5

Рисунок 46 – Таблица пользователей в базе данных PostgreSQL

*Шифрование:* при передаче данных между клиентом и сервером используется шифрование. Это обеспечивает защиту данных от перехвата и модификации в процессе передачи.

### 5.1.3 Безопасность в использовании SQL и резервное копирование

Параметризация SQL запросов: для обеспечения безопасности данных и предотвращения возможных SQL-инъекций, все SQL запросы в системе передаются в параметризованном виде. Этот подход обеспечивает, что данные, вводимые пользователем, обрабатываются таким образом, что они не могут быть интерпретированы как часть SQL команды. Это эффективно предотвращает атаки SQL-инъекции.

Например, в следующем коде выполняется проверка доступа пользователя. Здесь параметр "user\_id" передается в SQL запрос через метод AddWithValue. Это означает, что пользовательский ввод никогда не вставляется напрямую в SQL-запрос, что предотвращает возможность манипуляции с SQL-запросом.

```

Ссылка: 1
public bool CheckUserAccess(int userId)
{
    using (NpgsqlConnection connection = new NpgsqlConnection(connectionString))
    {
        connection.Open();

        using (NpgsqlCommand command = new NpgsqlCommand("SELECT COUNT(*) FROM права_доступа WHERE id_пользователя = @user_id", connection))
        {
            command.Parameters.AddWithValue("user_id", userId);

            int count = Convert.ToInt32(command.ExecuteScalar());

            if (count > 0)
            {
                return true; // пользователь имеет доступ к одной или нескольким базам данных
            }
            else
            {
                return false; // пользователь не имеет доступа к базам данных
            }
        }
    }
}

```

Рисунок 47 - Метод, использующий параметризованный SQL-запрос

*Резервное копирование:* Система поддерживает автоматическое резервное копирование данных. Это обеспечивает защиту от потери данных в результате сбоев системы или других непредвиденных обстоятельств. Благодаря хранению данных на сервере, возможность восстановления информации в случае ее потери увеличивается, что служит дополнительной мерой обеспечения целостности данных.

## 5.2 Безопасность и экологичность

В современном мире важность вопросов безопасности и экологии для разработки и эксплуатации информационных систем становится все более актуальной. Основные аспекты, на которые следует обратить внимание при создании и использовании системы управления общежитием, касаются безопасности рабочего места, учета экологических факторов и готовности к чрезвычайным ситуациям.

### 5.2.1 Безопасность

Кабинет коменданта общежития имеет размеры 2 метра на 2,5 метра с высотой помещения 3 метра. Окна помещения, ориентированные на север, оснащены металлопластиковыми рамами и стеклопакетами, а также дополнены вертикальными жалюзи для контроля освещения. Рабочее место коменданта оборудовано столом и офисным креслом для комфортного выполнения рабочих задач [7].

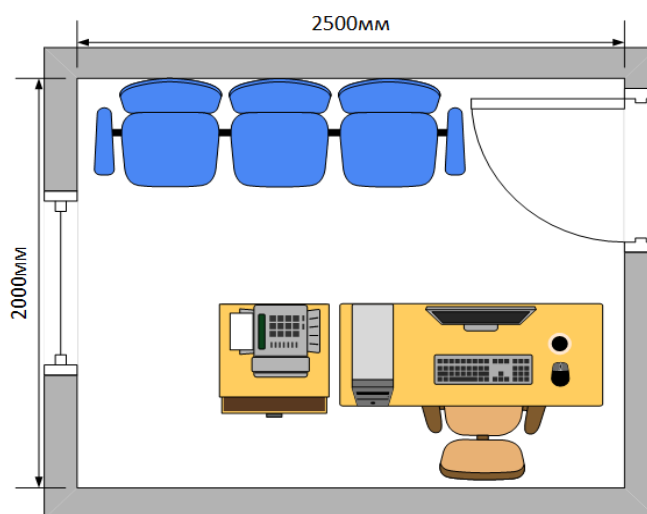


Рисунок 48 – Рабочее место коменданта общежития

Интерьер выполнен в темно-серых тонах, что обеспечивает неяркую, умеренную цветовую обстановку, способствующую концентрации.

Параметры микроклимата (температура, влажность воздуха, скорость движения воздуха) соответствуют санитарным нормам для помещений, в которых осуществляется умственный труд и легкая физическая работа категории 1а, согласно СанПиН 1.2.3685-21. Отопление, вентиляция и кондиционирование обеспечивают комфортные условия в течение всего года.

Таблица 35 - Таблица микроклимата для помещений с ЭВМ согласно СанПиН 1.2.3685-21

Категория работ	Температура воздуха (°С)			
	Диапазон ниже оптимальных величин		Диапазон выше оптимальных величин	
	Холодный период	Теплый период	Холодный период	Теплый период
1а	20,0 — 21,9	21,0 — 22,9	24,1 — 25,0	25,1 — 28,0
1б	19,0 — 20,9	20,0 — 21,9	23,1 — 24,0	24,1 — 28,0
2а	17,0 — 18,9	18,0 — 19,9	21,1 — 23,0	22,1 — 27,0
2б	15,0 — 16,9	16,0 — 18,9	19,1 — 22,0	21,1 — 27,0
3	13,0 — 15,9	15,0 — 17,9	18,1 — 21,0	20,1 — 26,0

Уровень шума и вибрации минимален. Источником шума служит лишь компьютер, уровень шума которого не превышает 35 дБ, что соответствует нормам для умственного труда [6].

Электромагнитное излучение от компьютерного оборудования также находится в пределах норм, установленных ГОСТ Р 50948-2001. Это обеспечивает безопасность использования электронного оборудования в рабочем пространстве.

Таблица 36 - нормируемые параметры электромагнитных излучений для рабочих мест с использованием ЭВМ в соответствии с ГОСТ Р 50948-2001

Частота	Напряжённость электрического поля, В/м, не более	Напряжённость магнитного поля, нТл, не более
---------	--	--

5 Гц – 2 КГц	25	250
2 КГц – 400 КГц	2.5	25

Электробезопасность. Источником напряжения питания является сеть переменного тока напряжением 220 В, к которой применяется ГОСТ 25861-83. В соответствии с требованиями «Стандартов безопасности при эксплуатации электрических систем потребителями» соблюдаются следующие требования по снижению вероятности возникновения аварийных ситуаций:

- электропитание ПЭВМ стабилизированно (с отклонением от 220В не более (10-15%)). Это делается путем подключения компьютера к источнику бесперебойного питания (АРС) и сетевому фильтру (Pilot);

- сетевое соединение для подключения устройств по всему зданию является трехпроводным: ноль электропитания, фаза, защитное заземление;

Освещение рабочего места коменданта общежития. Для обеспечения комфортных условий работы коменданта общежития, эффективности выполнения задач и минимизации утомления глаз, необходимо обеспечить правильное расположение рабочего места относительно источников освещения и грамотно рассчитать систему освещения.

Так же при работе с ПЭВМ могут возникнуть различные аварийные ситуации, например, пожар, взрыв, отравление персонала, в таких случаях необходимо организовать, так называемое, аварийное освещение или освещение безопасности. Освещение безопасности должно создавать на рабочих поверхностях в производственных помещениях и на территориях предприятий, требующих обслуживания при отключении рабочего освещения, наименьшую освещенность величиной 5 % освещенности, нормируемой для рабочего освещения от общего освещения, но не менее 2 лк внутри зданий и не менее 1 лк – для территорий предприятий [1].

Требования к освещению рабочего места коменданта включают:

- правильный выбор источников света и системы освещения, учитывающий не только их освещенность, но и тепловой комфорт и уровень бликов;
- обеспечение уровня освещенности, необходимого для рабочего стола и других поверхностей, на которых осуществляется работа;
- минимизация слепящего действия света, чтобы избежать утомления глаз и потери концентрации;
- ограничение или устранение колебаний светового потока, чтобы избежать создания неприятных условий и раздражения глаз;

В соответствии со стандартами гигиены минимальный уровень освещенности для рабочего места коменданта должен составлять 300 люкс на рабочей поверхности.

При работе коменданта общежития с ПЭВМ необходимо соблюдать требования норм и правил, установленные в соответствии с ГОСТ 12.0.003-2015 "Система стандартов безопасности труда. Электробезопасность. Общие требования". В этом стандарте указываются опасные и наносящие вред здоровью факторы при работе с ПЭВМ, которые включают: [2]

- повышенный уровень общей и локальной вибрации;
- повышенный уровень шума и другие неблагоприятные характеристики;
- повышенный уровень инфразвуковых и ультразвуковых колебаний;
- образование электростатических зарядов и наличие электростатического поля;
- наличие постоянного магнитного поля и электромагнитных полей промышленных и радиочастотных диапазонов;
- отсутствие или недостаток необходимого естественного и искусственного освещения;
- повышенная или пониженная температура воздуха, влажность и выделение химических веществ в воздух рабочей зоны;
- утомляемость глаз, монотонность трудового процесса и нервно-эмоциональные перегрузки;

- радиоактивное загрязнение поверхностей и материалов производственной среды;

Для предотвращения или снижения действия данных факторов необходимо принять меры и требования, описанные в ГОСТ 12.0.003-2015, а также разработать рекомендации для пользователей ПЭВМ, направленные на обеспечение безопасной работы и защиту здоровья;

В соответствии с требованиями к рабочему месту, оборудованному ПЭВМ, предъявляются следующие требования: [4]

- площадь на одно рабочее место сотрудника, проводящего за компьютером более четырех часов в день, должна составлять не менее 6 кв. м (если у компьютера монитор на базе электронно-лучевой трубки) или 4,5 кв. м (если монитор жидкокристаллический или плазменный);

- высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах (680 – 800) мм; при отсутствии такой возможности высота рабочей поверхности должна составлять 725 мм.

- рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной не менее 500 мм, глубиной на уровне колен не менее 450 мм и на уровне вытянутых ног не менее 650 мм.

- поверхность сиденья должна иметь ширину и глубину не менее 400 мм, иметь закругленный передний край, регулироваться в пределах (400 – 550) мм и углами наклона вперед до 15 градусов и назад до 5 градусов, а угол наклона спинки в вертикальной плоскости должен обеспечивать  $\pm 30$  градусов.

- стационарные или съемные подлокотники сиденья должны иметь длину не менее 250 мм и ширину (50 – 70) мм, регулироваться над сиденьем в пределах (230 $\pm$ 30) мм и внутреннего расстояния между подлокотниками в пределах (350 – 500) мм.

- рабочее место пользователя ПЭВМ должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах 150 мм и по углу наклона опорной поверхности подставки до 20 градусов.

– клавиатура должна располагаться на поверхности стола на расстоянии (100 – 300) мм от края, обращенного к пользователю, или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Рекомендуемое размещение пользователя ПЭВМ представлено на рисунке 48, где продемонстрировано правильное расположение монитора, клавиатуры, мыши и других элементов рабочего места.

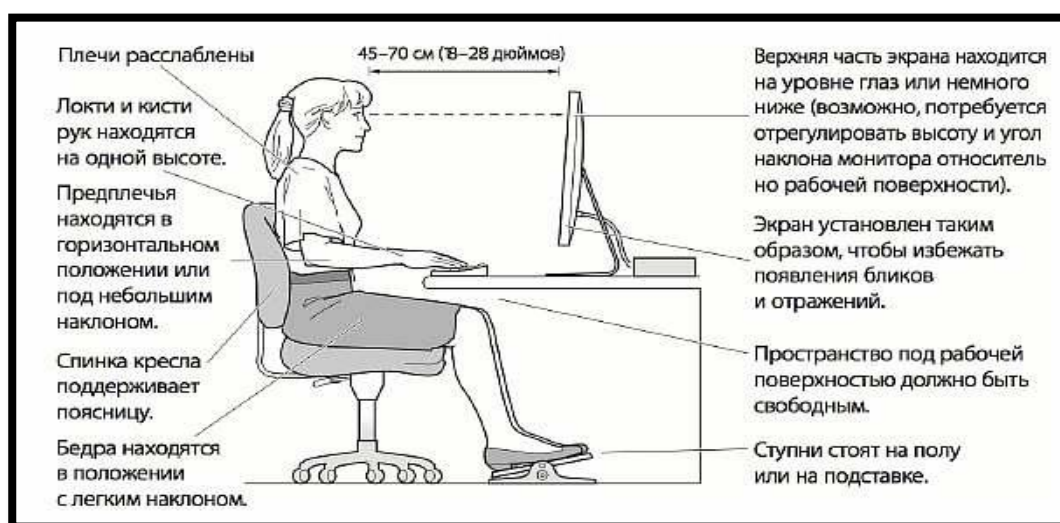


Рисунок 49 - Рекомендуемое размещение пользователя ПЭВМ

### 5.2.2 Экологичность

Внедрение автоматизированной системы управления общежитием ФГБОУ ВО "АмГУ" влечет за собой учет не только технологических, но и экологических аспектов. Особое внимание следует уделить правовым аспектам экологической безопасности.

В соответствии с Федеральным законом "Об отходах производства и потребления" от 24 июня 1998 г. № 89-ФЗ, важно обеспечить надлежащую утилизацию отходов, включая устаревшее компьютерное и офисное оборудование. Это оборудование может содержать в себе вредные вещества, которые при неправильной утилизации могут нанести вред окружающей среде.

Кроме того, в соответствии с Федеральным законом "О драгоценных металлах и драгоценных камнях" от 26 марта 1998 г. № 41-ФЗ, списанное компьютерное оборудование, содержащее драгоценные металлы, подлежит специальному учету и утилизации.

Также следует отметить, что внедрение автоматизированной системы может способствовать снижению негативного воздействия на окружающую среду. Применение цифровых технологий позволяет уменьшить потребление бумаги и сократить объем производимых отходов.

В Благовещенске, где находится университет, есть ряд компаний, специализирующихся на утилизации компьютеров, оргтехники и электроники. Среди них:

- РОСА-1;
- ЭкоАрхитектура;
- ЛегионСтрой;
- ЦУТО;
- ИП Виноградова О Г - представитель ООО Ведущая Утилизирующая Компания;
- Русутилит;

Также в городе работает компания "EcoLife", которая занимается утилизацией и переработкой различных видов отходов, включая электронику

В целом, несмотря на возможные экологические риски, связанные с использованием и утилизацией электронного оборудования, внедрение автоматизированной системы управления может привести к значительным экологическим преимуществам при соблюдении всех необходимых экологических и правовых норм.

### **5.2.3 Чрезвычайные ситуации**

Чрезвычайные ситуации являются непредвиденными событиями, которые могут возникнуть в организации и представлять угрозу для безопасности и здоровья сотрудников. Они могут быть вызваны различными факторами, такими как пожар, наводнение, авария на оборудовании и другие.



Для обеспечения безопасности и эффективного реагирования на чрезвычайные ситуации необходимо разработать и внедрить соответствующие меры и процедуры. Важными аспектами в этом процессе являются:

*Планирование и подготовка:*

- разработка плана чрезвычайных ситуаций, включающего процедуры эвакуации, сбора и передачи информации, оказания первой помощи и другие необходимые шаги;
- организация тренировок и учений сотрудников для проверки их готовности к действиям в чрезвычайных ситуациях;
- обеспечение наличия необходимого оборудования и ресурсов для реагирования на чрезвычайные ситуации, таких как огнетушители, аварийные выходы, медицинские принадлежности;

*Профилактика и предупреждение:*

- регулярное проведение проверок и обслуживания оборудования, систем безопасности и противопожарных средств;
- внедрение мер по предупреждению чрезвычайных ситуаций, таких как контроль за использованием электрооборудования, обучение сотрудников правилам пожарной безопасности;

*Быстрое реагирование и эвакуация:*

- обеспечение оперативной реакции на чрезвычайные ситуации, включая активацию сигнализации и систем оповещения;
- организация эвакуации сотрудников и посетителей с учетом опасности и безопасных маршрутов;
- обеспечение доступа к местам скрытого пожара и применение средств пожаротушения при необходимости;

*Последующие действия:*

- проведение анализа причин и последствий чрезвычайной ситуации и разработка мер для предотвращения подобных инцидентов в будущем;
- восстановление работоспособности организации и ее инфраструктуры после чрезвычайной ситуации;

- проведение обучения и информирования сотрудников о произошедшем инциденте и принятых мерах;

Требования по обеспечению пожарной безопасности. Работник должен:

- осведомляться о плане эвакуации и расположении огнетушителей;
- знать, как правильно использовать огнетушители;
- не блокировать проходы посторонними предметами;
- выключать компьютер при длительных перерывах и уходе с работы;
- избегать размещения горючих материалов (тканей, бумаги и т.д.) рядом с настольными лампами;
- спокойно и без паники покидать здание при общем сигнале опасности;
- курить только в специально отведенных местах;
- в случае пожара прекратить работу, отключить электрооборудование, предупредить коллег и, если возможно, вызвать пожарных по номеру «01» и потушить пожар с использованием имеющихся средств пожаротушения;

*Работнику запрещено:*

- использовать открытый огонь или курить в рабочем кабинете;
- хранить легковоспламеняющиеся вещества в рабочем кабинете;
- оставлять без присмотра неисправное электрооборудование (компьютеры, обогреватели и т.д.);
- использовать неисправные электрические устройства;

*Требования пожарной безопасности к помещениям:*

- противопожарные системы и установки (системы защиты от дыма, пожарная автоматизация) в помещениях, зданиях и сооружениях должны всегда находиться в хорошем техническом состоянии;
- напольные покрытия, такие как ковры, должны быть надежно закреплены к полу в помещениях с массовым пребыванием людей;
- эвакуационные маршруты и выходы не должны быть загромождены различными материалами, продуктами, устройствами, отходами и другими предметами, а также двери эвакуационных выходов не должны быть заблокированы;

Требования пожарной безопасности к оборудованию и электроустановкам:

- электрические установки и бытовые приборы в помещениях, где нет обслуживающего персонала в конце рабочего дня, должны быть отключены;

Однако аварийные, пожарные и охранные сигнализации должны оставаться активными. В случае других электрических установок, их можно оставить под напряжением, если это связано с их функциональным назначением или предусмотрено в требованиях руководства пользователя.

- запрещается использовать поврежденные розетки, автоматические выключатели. Также необходимо избегать обертывания электроламп и светильников бумагой, тканью и другими горючими материалами, а также эксплуатации светильников без колпаков (рассеивателей), которые предусмотрены конструкцией светильника;

## ЗАКЛЮЧЕНИЕ

Разработанная автоматизированная система управления общежитием для ФГБОУ ВО «АмГУ» на основе PostgreSQL и формах WPF является неотъемлемым инструментом в текущих условиях пандемии COVID-19 и связанных с ней ограничений на прямое общение. Эта система способствует оптимизации и усовершенствованию управленческих процессов в общежитиях, улучшает условия проживания и обеспечивает безопасность жителей.

Созданная на базе PostgreSQL, система управления общежитием была специально адаптирована под потребности комендантов общежитий, а также оснащена мощными механизмами защиты данных, в полном соответствии с нормами информационной безопасности

Ожидается, что использование этой системы значительно улучшит процесс управления общежитием, а также поможет сократить затраты на приобретение и техническую поддержку программного обеспечения в будущем. Это позволит управляющим структурам сосредоточиться на первостепенных задачах, а не на решении технических вопросов, увеличивая эффективность и качество управления.

## БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

- 1 Виды и системы освещения [Электронный ресурс]. Доступно по адресу: <http://electricalschool.info/main/lighting/1939-vidy-i-sistemy-osveshhenija.html>
- 2 ГОСТ 12.0.003-2015. Система стандартов безопасности труда. Опасные и вредные производственные факторы. – Введ. 2017-03-01. – МКС: Межгосударственный совет по стандартизации, метрологии и сертификации; М. : Изд-во стандартов, 2015.
- 3 ГОСТ Р 50948-2001 "Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности".
- 4 Куренкова, Г. В. Гигиенические особенности условий труда и здоровье профессиональных пользователей персональных компьютеров и видеодисплейных терминалов / Г. В. Куренкова // Сиб. мед. журн. - 2004. - № 6. - С. 14-17.
- 5 Основы безопасности жизнедеятельности [Электронный ресурс]. Доступно по адресу: <https://resh.edu.ru/subject/lesson/5829/train/104190/>
- 6 СанПин 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания".
- 7 Тарасенко, Н.Ю. и Волкова, З.А. "Профессиональные вредности" // Большая медицинская энциклопедия: в 30 т. / гл. ред. Б.В. Петровский. – 3 изд. – Москва: Советская энциклопедия, 1983. – Т. 21. Преднизон - Растворимость. – 560 с.
- 8 Федеральный закон от 30.03.1999 № 52-ФЗ "О санитарно-эпидемиологическом благополучии населения".

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Биль, Е. А. Проектирование автоматизированной информационной системы «Студгородок» (на примере управления студенческого городка ФГБОУ ВО «Алтайский государственный технический университет им. И. И. Ползунова) / Е. А. Биль // Наука и молодежь : Материалы XVIII Всероссийской научно-технической конференции студентов, аспирантов и молодых ученых. В 2-х томах, Барнаул, 19–23 апреля 2021 года / Отв. редактор М.В. Гунер. Том 1. Часть 1. – Барнаул: Алтайский государственный технический университет им. И.И. Ползунова, 2021. – С. 20-21. – EDN IGOVDS.
- 2 ГОСТ 34.601-90 Автоматизированные системы. Стадии создания. – Взамен ГОСТ 24.601-86, ГОСТ 24.602-86; введ. 1992–01–01. – М. : Изд-во Стандартиформ, 2009. – 6 с.
- 3 ГОСТ 34.602-89. Техническое задание на создание автоматизированной системы. – Взамен ГОСТ 24.201-85; введ. 1990–01–01. – М. : Изд-во Стандартиформ, 2009. – 12 с.
- 4 ГОСТ Р ИСО/МЭК 27001-2013 "Информационная технология. Методы обеспечения безопасности. Системы управления информационной безопасностью. Требования.
- 5 Стандарт организации. Оформление выпускных квалификационных и курсовых работ (проектов) [Электронный ресурс] / АмГУ ; разработ. Л. А. Проказина, Н. А. Чалкина, С. Г. Самохвалова. – Введ. с 05.04.2018. – Благовещенск : [б. и.], 2018. – 75 с.
- 6 Терентьева, А. П. Разработка информационной системы "Учет студентов, проживающих в общежитиях ФГБОУ ВО "УГТУ" / А. П. Терентьева, С. Г. Куделин // Информационные технологии в управлении и экономике. – 2018. – № 2(11). – С. 35-43. – EDN WQEQHO.
- 7 Юркова, О. Н. Разработка специального программного обеспечения систем управления и механизмов принятия решений для управления студенческими общежитиями / О. Н. Юркова, М. А. Слепцова // Современная наука:

актуальные проблемы теории и практики. Серия: Естественные и технические науки. – 2019. – № 12. – С. 100-105. – EDN FLHCIQ.

8 Clarke, Richard A., and Knake, Robert K. "The Fifth Domain: Defending Our Country, Our Companies, and Ourselves in the Age of Cyber Threats". Penguin Random House, 2019. Доступно по ссылке: <https://www.penguinrandomhouse.com/books/586684/the-fifth-domain-by-richard-a-clarke-and-robert-k-knake/>

9 Diogenes, Yuri, and Ozkaya, Erdal. "Cybersecurity – Attack and Defense Strategies". Packt Publishing, 2018. Доступно по ссылке: <https://www.packtpub.com/product/cybersecurity-attack-and-defense-strategies/9781788475297>

10 Menn, Joseph. "Cult of the Dead Cow: How the Original Hacking Supergroup Might Just Save the World". PublicAffairs, 2019. Доступно по ссылке: <https://www.amazon.com/Cult-Dead-Cow-Original-Supergroup/dp/154176238X>

11 Steinberg, Joseph. "Cybersecurity For Dummies". For Dummies, 2019. Доступно по ссылке: <https://www.amazon.com/Cybersecurity-Dummies-Joseph-Steinberg/dp/1119560322>

12 Stuttard, Dafydd, and Pinto, Marcus. "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws". Wiley, 2018. Доступно по ссылке: <https://www.wiley.com/enus/The+Web+Application+Hacker%27s+Handbook%3A+Finding+and+Exploiting+Security+Flaws%2C+2nd+Edition-p-9781118026472>

13 ZetCode. (2023, January 4). ADO.NET - PostgreSQL. Доступно по ссылке: <http://zetcode.com/csharp/postgresql/>

14 ZetCode. (2023, January 4). C# PostgreSQL - programming PostgreSQL in C#. Доступно по ссылке: <http://zetcode.com/csharp/postgresql/>

15 ZetCode. (2023, January 4). Working with PostgreSQL in C#. Доступно по ссылке: <http://zetcode.com/csharp/postgresql/>

# ПРИЛОЖЕНИЕ А

## ТЕХНИЧЕСКОЕ ЗАДАНИЕ

«Разработка автоматизированной системы управления общежитием для ФГБОУ ВО «АмГУ»»

### 1 ОБЩИЕ СВЕДЕНИЯ

#### 1.1 Полное наименование системы

Полное наименование разрабатываемой автоматизированной системы управления – Автоматизированная система управления общежитием для ФГБОУ ВО «АмГУ».

#### 1.2 Разработчик

Разработчик – студент 953-об группы факультета математики и информатики Амурского государственного университета – Яковлев Максим Дмитриевич.

#### 1.3 Перечень документов

Перечень документов, на основе которых разрабатывается система:

- ГОСТ 19.001-77 – общие положения;
- ГОСТ 19.004-80 – термины и определения;
- ГОСТ 19.101-77 – виды программ и программных документов;
- ГОСТ 19.102-77 – стадии разработки;
- ГОСТ 19.103-77 – обозначение программ и программных документов;
- ГОСТ 19.104-78 – основные надписи;
- ГОСТ 19.105-78 – общие требования к программным документам;
- ГОСТ 19.106-78 – требования к программным документам, выполненным печатным способом;
- ГОСТ 19.402-78 – описание программы;
- ГОСТ 19.502-78 – описание применения. Требования к содержанию и оформлению;



## Продолжение приложения А

- ГОСТ 19.505-79 – руководство оператора. Требования к содержанию и оформлению;
- ГОСТ 19.508-79 – руководство по техническому обслуживанию. Требования к содержанию и оформлению;
- ГОСТ 34.602-89 – техническое задание на создание автоматизированной системы;
- ГОСТ 34.201-89 – виды, комплектность и обозначение документов при создании автоматизированных систем;
- ГОСТ 24.104-85 – автоматизированные системы управления. Общие требования;
- ГОСТ 34.601-90 – автоматизированные системы. Стадии создания;
- ГОСТ 25.861-83 – АСУ. Требования по безопасности средств вычислительной техники;
- инструкция по охране труда при работе на персональном компьютере;
- первичные документы;
- требования к системе;
- должностные инструкции;

### **1.4 Плановые сроки начала и окончания работы**

Плановые сроки начала и окончания работ по созданию системы: начало разработки - 30.01.2023 г., окончание – 20.06.2023 г.

## **2 НАЗНАЧЕНИЕ И ЦЕЛИ СОЗДАНИЯ СИСТЕМЫ**

### **2.1 Назначение системы**

Разрабатываемая система содержит следующие функции:

- управление анкетными данными и данными об оплате за проживание студентов;

## Продолжение приложения А

- оповещение студентов о задолженности за проживание в автоматическом и ручном режимах;
- поиск должников с помощью применения фильтров поиска;
- осуществление заселения, переселения и выселения студентов из комнат с указанием причины, а также добавления используемых электроустановок к студенту при заселении;
- управление данными о комнатах и вызова сотрудников в комнаты для устранения сантехнических, электрических неисправностей и других проблем;
- организация плана мероприятий в общежитии и оповещения коменданта о запланированных мероприятиях;
- генерирование отчетов по оплате за проживание, должникам, заселению/выселению, комнатам, мероприятиям и доступности комнат для заселения в формате документа Excel;

### **2.2 Цели создания системы**

Целью создания автоматизированной системы управления общежитиями является упрощение процесса учета студентов и их поселения, увеличение прозрачности и эффективности управления общежитием, обеспечение быстрого и удобного доступа к информации, сокращение времени на выполнение административных задач, минимизация рисков ошибок при ведении учета и обработке данных, создание возможности для автоматического формирования отчетов и расширения функциональности системы.

### **3 ХАРАКТЕРИСТИКИ ОБЪЕКТА АВТОМАТИЗАЦИИ**

Объектом разработки системы являются общежития ФГБОУ ВО «АмГУ».

Программное обеспечение предназначено для автоматизации и упрощения работы с базой данных общежитий ФГБОУ ВО «АмГУ». Данная система позволит заведующему общежитием наиболее эффективно управлять потоком информации в своём общежитии.

## 4 ТРЕБОВАНИЯ К СИСТЕМЕ

### 4.1 Требования к системе в целом

Требования:

- безопасность данных: система должна обеспечивать сохранность и конфиденциальность персональных данных студентов, а также защиту системы от несанкционированного доступа;
- удобный и интуитивно понятный интерфейс: система должна иметь простой и понятный интерфейс для удобства использования пользователем;
- оптимизация процессов: система должна оптимизировать процессы управления общежитием, включая управление проживающими, комнатами, оплатой за проживание и планированием мероприятий;
- автоматизация: система должна иметь возможность автоматического уведомления студентов о задолженностях и запланированных мероприятиях, а также генерации отчетов;
- гибкость: система должна иметь возможность настройки и адаптации к особенностям конкретного общежития;

### 4.2 Требования к структуре и функционированию системы

#### 4.2.1 Перечень подсистем, их назначение

Проектируемая система будет представлена смежными вложенными в нее подсистемами:

- подсистема ввода данных, представленная понятным для восприятия и удобным для работы интерфейсом;
- подсистема хранения данных, представленная в виде физических таблиц данных, которые будут получены после выполнения всех этапов проектирования базы данных (изучение предметной области, инфологическое, логическое и физическое проектирование), а затем будет реализована в СУБД;

## Продолжение приложения А

- подсистема вывода данных, которая позволяет компоновать результирующие данные в удобной для пользователя форме;

### **4.2.2 Перспективы развития, модернизация системы**

При разработке системы должны быть предусмотрены возможности ее последующей модернизации при минимальных временных и финансовых затратах по следующим направлениям:

- расширение прикладных функций;
- изменение интерфейса системы;
- изменение форматов и протоколов обмена данных;

### **4.2.3 Требования к численности и квалификации персонала**

Проектируемая система не накладывает ограничений на численность персонала и предназначена для специалистов с базовыми навыками работы на персональном компьютере.

В соответствии с правами доступа, пользователей можно разделить на 2 группы:

- заведующий общежитием;
- администратор;

Заведующий имеет доступ только к той части, которая находится в области для управления процессами в общежитии.

Администратор имеет доступ к административной панели приложения, а также имеет доступ ко всем базам данных общежитий при авторизации в программе.

Доступ к административной части должен осуществляться с использованием уникального логина и пароля.

### **4.2.4 Требования к показателям назначения**

Система должна иметь понятный интерфейс.

Целевое назначение системы должно сохраняться на протяжении всего срока эксплуатации.

## Продолжение приложения А

### **4.2.5 Требования к надежности**

Программа должна отвечать следующим требованиям надежности:

- защита от некорректных действий пользователя, которые могут привести к сбою в программе;
- контроль операций, анализ результатов на наличие ошибок, выявление причины ошибок, исправление ошибок;

### **4.2.6 Требования к безопасности**

Требования:

- проверка данных на достоверность;
- обеспечение безопасного хранения данных;
- обеспечение безопасного режима передачи данных;
- расположение информации в закрытой БД, доступ к которой разрешен только с использованием пароля доступа;
- предотвращение действий, которые могут привести к разрушению, искажению, уничтожению информации или сбоев в работе средств автоматизации;

### **4.2.7 Требования к эргономике и технической эстетики**

Создаваемая система должна отвечать требованиям эргономики, то есть быть максимально понятной и обеспечивать комфортную работу пользователя в самой системе. Система должна обеспечивать максимальную скорость ввода информации. Интерфейс системы должен быть максимально понятным и акцентировать внимание пользователя.

### **4.2.8 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению**

Пользователи должны быть ознакомлены с правилами эксплуатации всех технических средств и регламентом работы системы. Качественная работа

## Продолжение приложения А

системы, обеспечивается соблюдением всех требований эксплуатационных документов.

Устройство хранения данных должно быть защищено от внешних физических воздействий. Для надежного хранения данных, будет разработана система разграничения прав доступа между пользователями, а также предусмотрена система паролей.

### **4.2.9 Требования к сохранности информации при авариях**

Данные требования заключаются в сохранности информации в случае возникновения программных и технических сбоев, а также сбоев операционной системы и допущение ошибок пользователями при работе в системе.

Специализированные программные средства администратора системы должны обеспечивать:

- мгновенное восстановление информации;
- сохранение информации при аварийных ситуациях, а в случае ее потери – возможность полного или частичного восстановления информации;
- в случае выхода из строя технического средства должна обеспечиваться его замена без потери функциональности подсистемы;
- наличие инструкций при возникновении аварийных ситуаций;
- сохранение резервной копии на носителе;

### **4.2.10 Требования к защите информации от несанкционированного доступа**

Система должна иметь защиту от несанкционированного дублирования и переноса данных на другой компьютер, и для каждого пользователя необходимо задать пароль и права доступа к данным.

## **4.3 Требования к видам обеспечения**

### **4.3.1 Требования к информационному обеспечению**

## Продолжение приложения А

Проектируемая автоматизированная система управления общежитием должна содержать следующие данные:

- список студентов, проживающих в общежитии, включая данные о каждом студенте (ФИО, группа, контактная информация и прочее);
- информацию об условиях проживания в общежитии (количество комнат, количество мест в каждой комнате, наличие санитарных узлов, кухонь и прочее);
- расписание работы общежития, включая часы работы администрации, график уборки, проведения ремонтных работ и прочее;
- информацию о правилах проживания в общежитии, включая требования к поведению студентов, порядок выдачи ключей от комнат, порядок оплаты проживания и прочее;
- данные о наличии свободных мест в общежитии и процедуре заселения студентов;
- отчеты об учете проживающих студентов, включая информацию о численности студентов, информацию об их оплате, данных о выселении и прочее;
- документацию, необходимую для управления общежитием, включая договоры на аренду помещений, акты о приеме-передаче помещений, акты выполненных работ и прочее;

### **4.3.2 Требования к программному обеспечению**

Требования к программному обеспечению заключается в выборе платформы для разрабатываемой системы.

Система совместима со всеми версиями ОС Microsoft Windows.

В качестве СУБД будет использоваться СУБД PostgreSQL. Эта СУБД реализует архитектуру клиент-сервер, обеспечивает надежную защиту данных, возможность работы в многопользовательском режиме.

## Продолжение приложения А

В качестве среды разработки был выбран Microsoft Visual Studio 2022.

### **4.3.3 Требования к техническому обеспечению (аппаратные ограничения)**

Требования к рабочим станциям должны быть минимальными и обеспечивать функционирование системы без сбоев и ошибок:

- процессор (Intel или AMD) от 1 ГГц;
- объем оперативной памяти более 512 Мб;
- монитор;
- устройства ввода информации: клавиатура, мышь;
- сетевая карта с пропускной способностью от 100 Мбит/сек;

Данные характеристики были выбраны для эффективной работы без ожидания отклика системы на запросы, а также для обеспечения целостности и сохранности информации при сбоях.

### **4.3.4 Требования к лингвистическому обеспечению.**

Все прикладное программное обеспечение системы для организации взаимодействия с пользователем должно использовать русский язык.

При реализации системы должны применяться следующие языки программирования: SQL, C#.

## **5 СОСТАВ И СОДЕРЖАНИЕ РАБОТ ПО СОЗДАНИЮ СИСТЕМЫ**

Этапы, которые необходимо выполнить при создании информационной системы:

1 этап – Исследование предметной области, выделение объекта автоматизации, анализ деятельности предприятия;

2 этап – Составление технического задания: выявление пожеланий заказчика к разрабатываемой системе, определение технических и программных средств, для реализации проекта;

3 этап – Разработка автоматизированной системы: физическое, логическое и инфологическое проектирование системы;



## Продолжение приложения А

4 этап – Реализация автоматизированной системы;

5 этап – Согласование автоматизированной системы с требованиями заказчика, учет всех пожеланий и замечаний;

### 6 ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ СИСТЕМЫ

При приеме автоматизированной системы управления заказчик должен ознакомиться с документацией и руководством пользователя. Прием промежуточных и окончательных работ должен проводиться с участием непосредственно тех лиц, которые будут работать с данной информационной системой.

Заказчик должен проверить систему на соответствие предъявляемым требованиям.

База данных должна содержать необходимые данные для проведения тестирования. Все тесты проводятся в условиях реальной работы. Результаты тестов должны соответствовать требованиям, предъявляемым к системе.

### 7 ТРЕБОВАНИЯ К СОСТАВУ И СОДЕРЖАНИЮ РАБОТ ПО ПОДГОТОВКЕ ОБЪЕКТА АВТОМАТИЗАЦИИ К ВВОДУ СИСТЕМЫ В ДЕЙСТВИЕ

Основные мероприятия, необходимы для ввода системы в действие:

- обучение персонала работе с системой;
- изменения, которые необходимо осуществить в объекте автоматизации;
- создание условий функционирования системы, при которых гарантируется соответствие создаваемой системы требованиям, содержащимся в техническом задании;

Этапы, которые необходимо выполнить по созданию информационной системы:

- разработка БД системы;
- разработка функционала;
- разработка графического интерфейса системы;

## Продолжение приложения А

- тестирование;
- внедрение;

## ПРИЛОЖЕНИЕ Б

Окно авторизации

**Авторизация в Автоматизированной Системе Управления Общежитием**

Выберите своё общежитие из списка: Общежитие 2

Имя пользователя: admin

Пароль: ●●●●

Панель администратора

Авторизация

Logo of AmGU (Amurskiy Gosudarstvennyy Universitet) with the year 1975.

Рисунок Б.1 – Форма авторизации

Окно авторизации

**Авторизация в Автоматизированной Системе Управления Общежитием**

Выберите своё общежитие из списка: Общежитие 2

Логин: [input field]

Пароль: [input field]

Войти в окно администратора

Панель администратора

Авторизация

Logo of AmGU (Amurskiy Gosudarstvennyy Universitet) with the year 1975.

Background text from the main window: Имя пользователя: admin, Пароль: ●●●●

Рисунок Б.2 – Форма авторизации для входа в панель администратора

## Продолжение приложения Б

**Панель администратора**

Имя пользователя:

Пароль:

Подтверждение пароля:

Роль:

**Создать пользователя**

Имя пользователя	Роль
Zav1	Зав.общезитием №1
Zav2	Зав.общезитием №2
Zav3	Зав.общезитием №3
Zav3L	Зав.общезитием №3Л

**Удалить пользователя**

**Назначить права доступа**

**Выйти к авторизации**

Рисунок Б.3 – Панель администратора

**Назначение прав пользователю**

Выбор пользователя:

Имя пользователя	Роль	Доступ
Zav1	Зав.общезитием №1	Общезитие 1
Zav2	Зав.общезитием №2	Общезитие 2
Zav3	Зав.общезитием №3	Общезитие 3

Выбор общезития:

**Выдать доступ**

**Удалить права доступа**

Рисунок Б.4 – Окно администратора для выдачи прав доступа

## Продолжение приложения Б

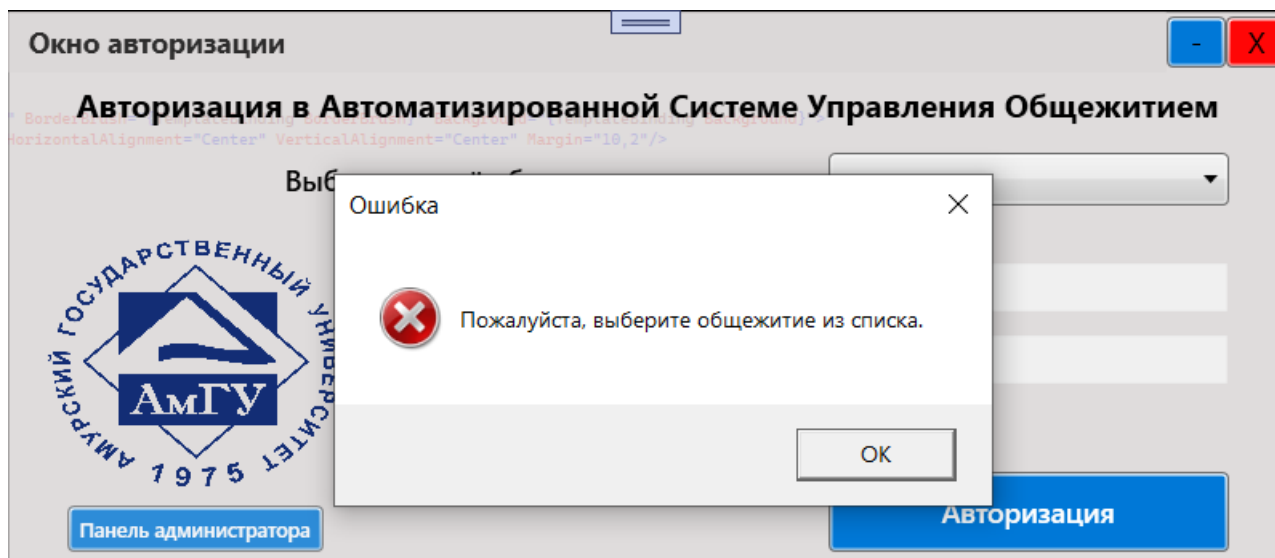


Рисунок Б.5 – Ошибка по причине не выбранного общежития при авторизации

## ПРИЛОЖЕНИЕ В

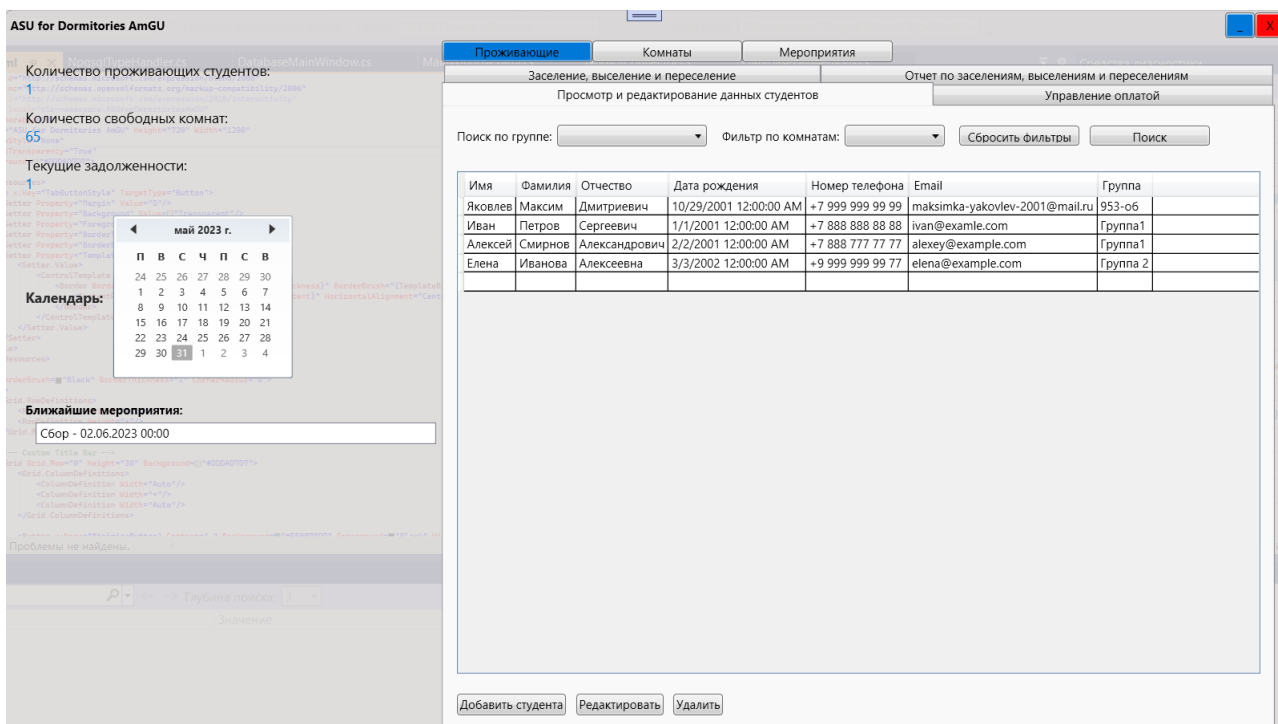


Рисунок В.1 – Раздел «Просмотр и редактирование данных студентов»

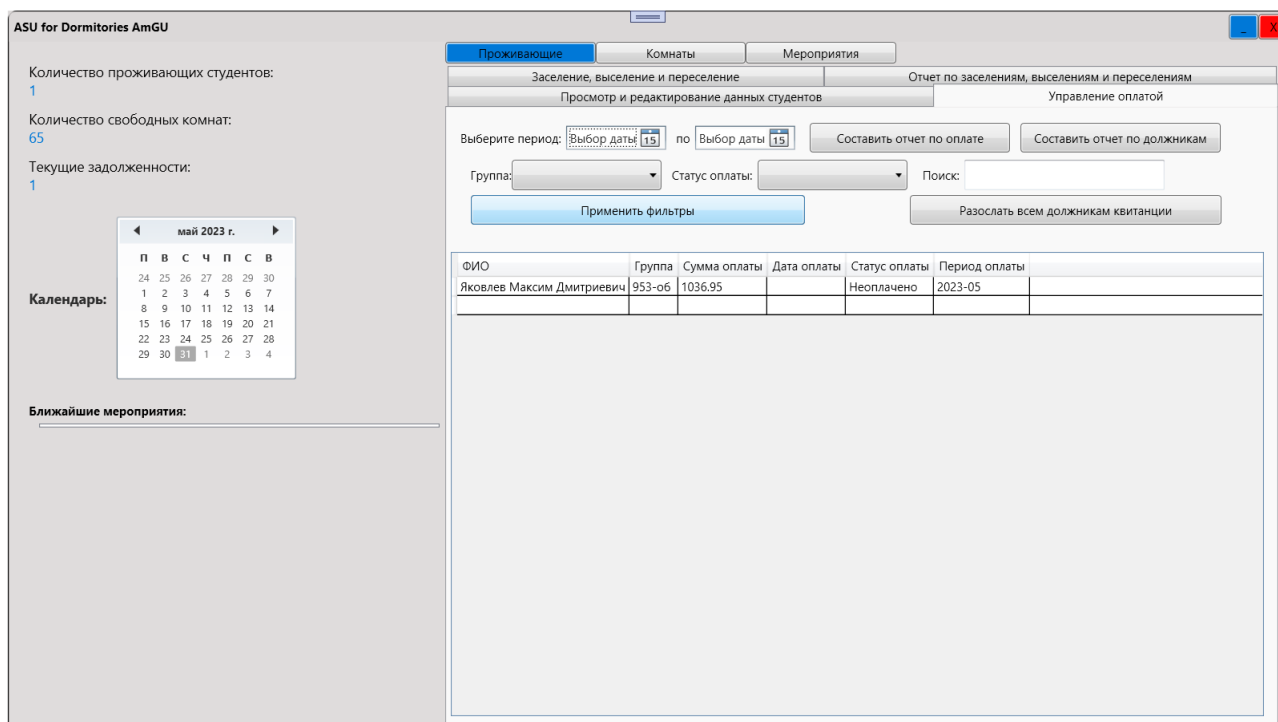


Рисунок В.2 – Раздел «Управление оплатой»

## Продолжение приложения В

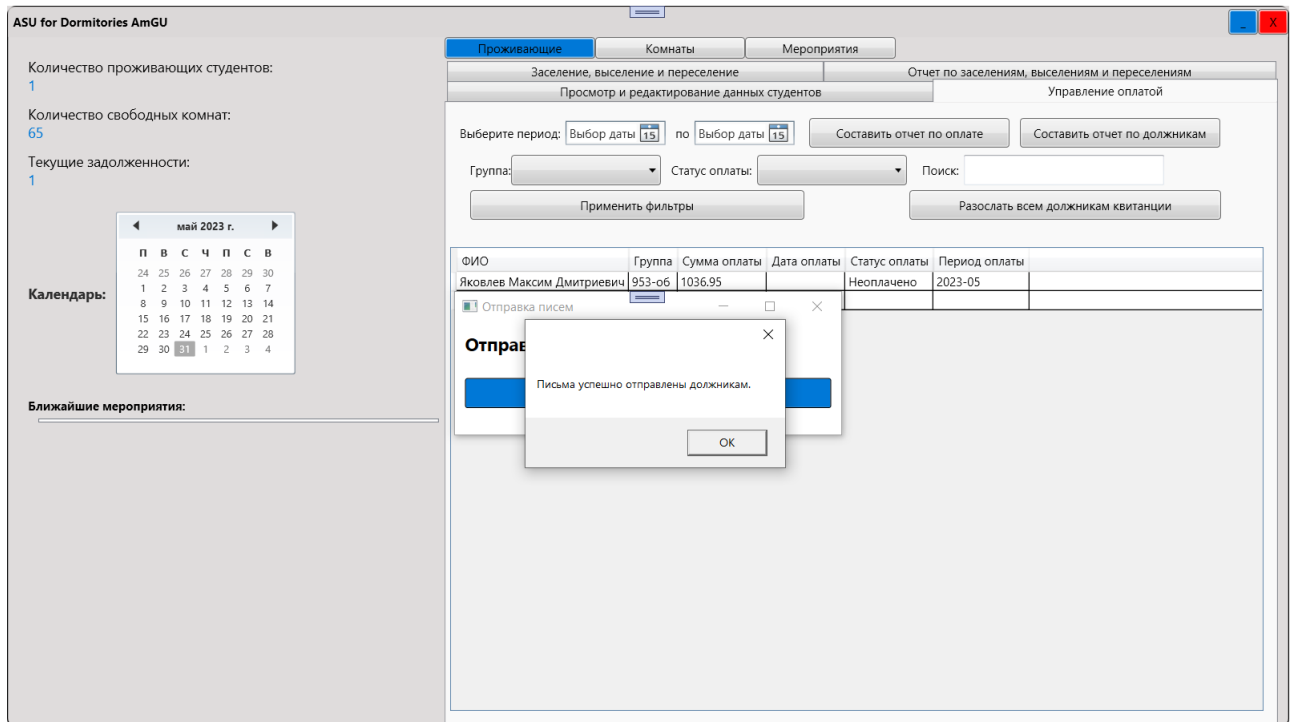


Рисунок В.3 – Процесс отправки писем о задолженности студентам

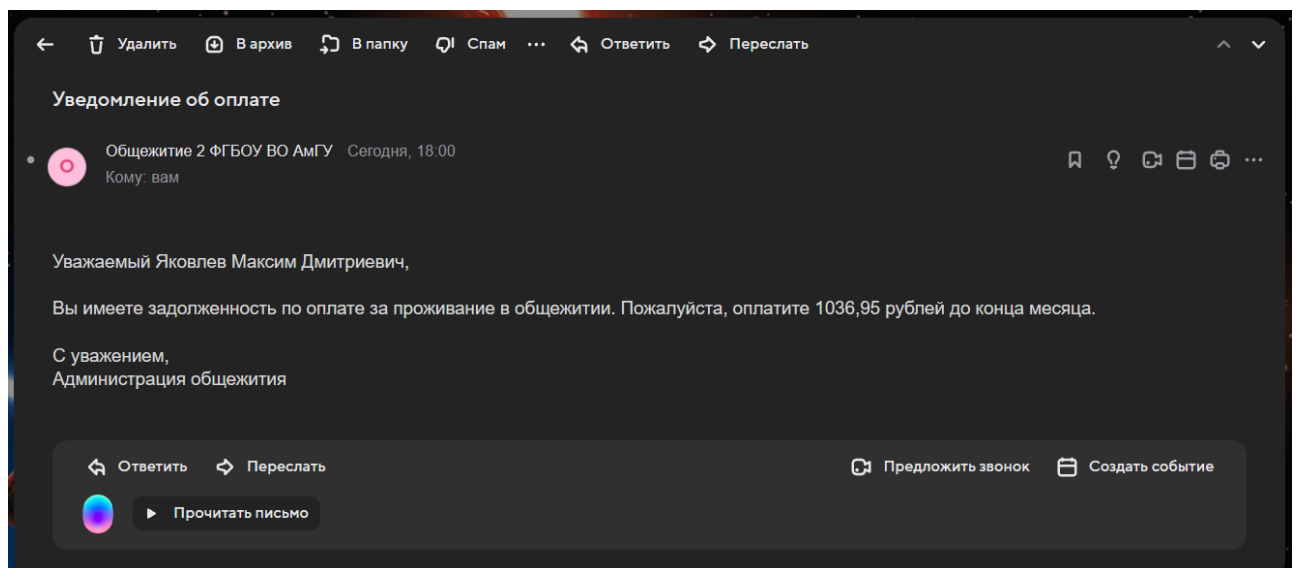


Рисунок В.4 – Пришедшее письмо на электронную почту о задолженности

## Продолжение приложения В

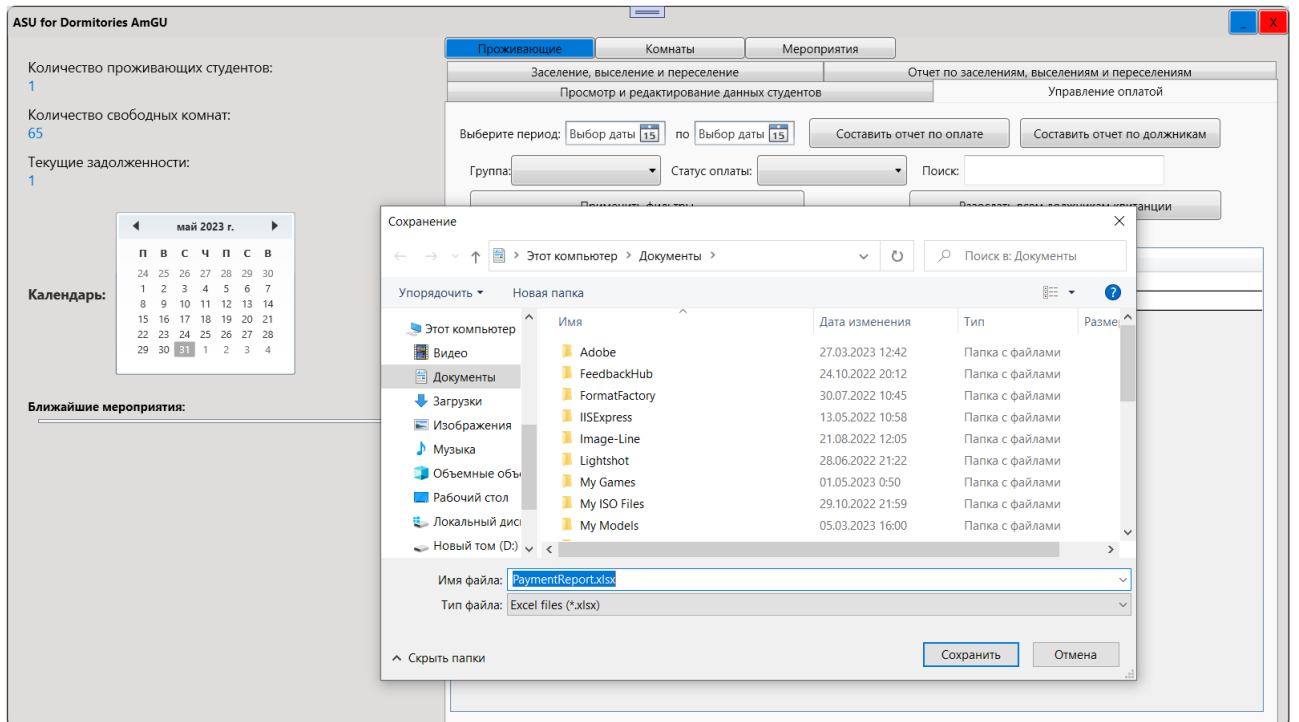


Рисунок В.5 – Процесс сохранения отчета по оплате в документ Excel

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I
1	ФИО	Группа	Сумма оплаты	Дата оплаты	Статус оплаты	Период оплаты			
2	Яковлев Максим Дмитриевич	953-об	1036,95	-	Неоплачено	2023-05			

Рисунок В.6 – Сохраненный документ Excel с отчётом по оплате



## ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ В

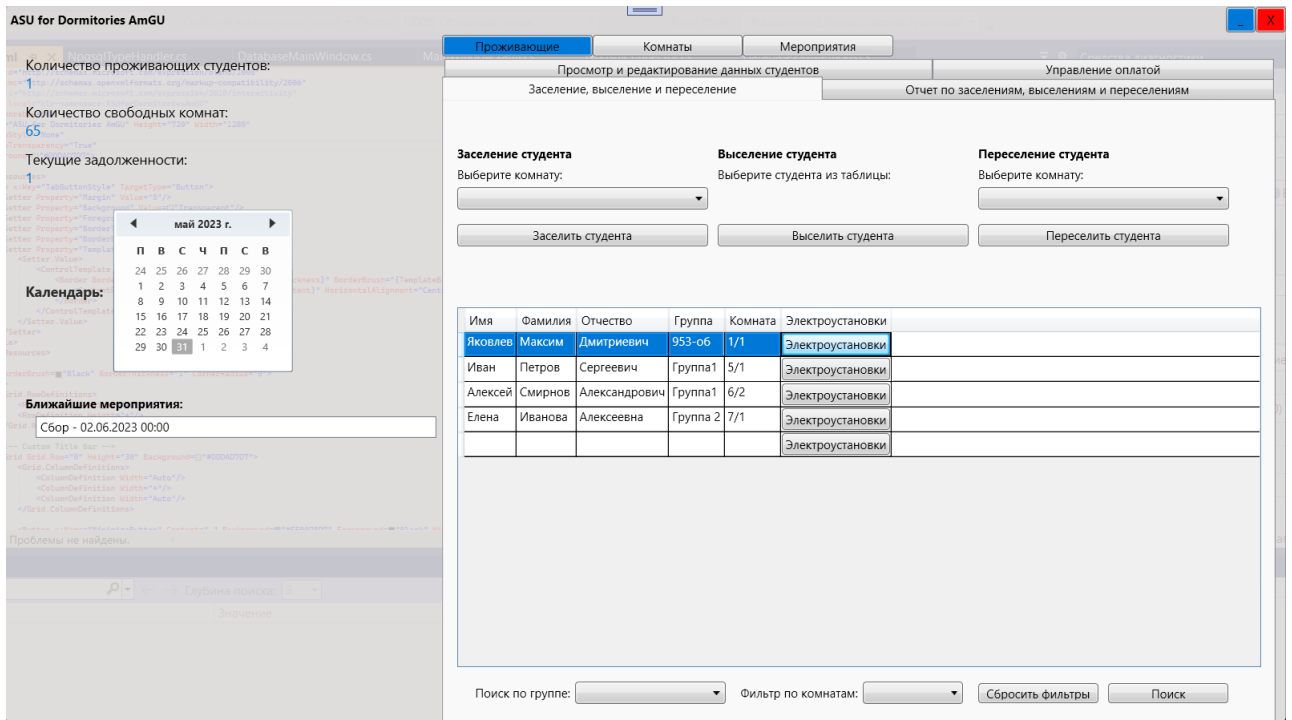


Рисунок В.7 – Раздел «Заселение, выселение и переселение»

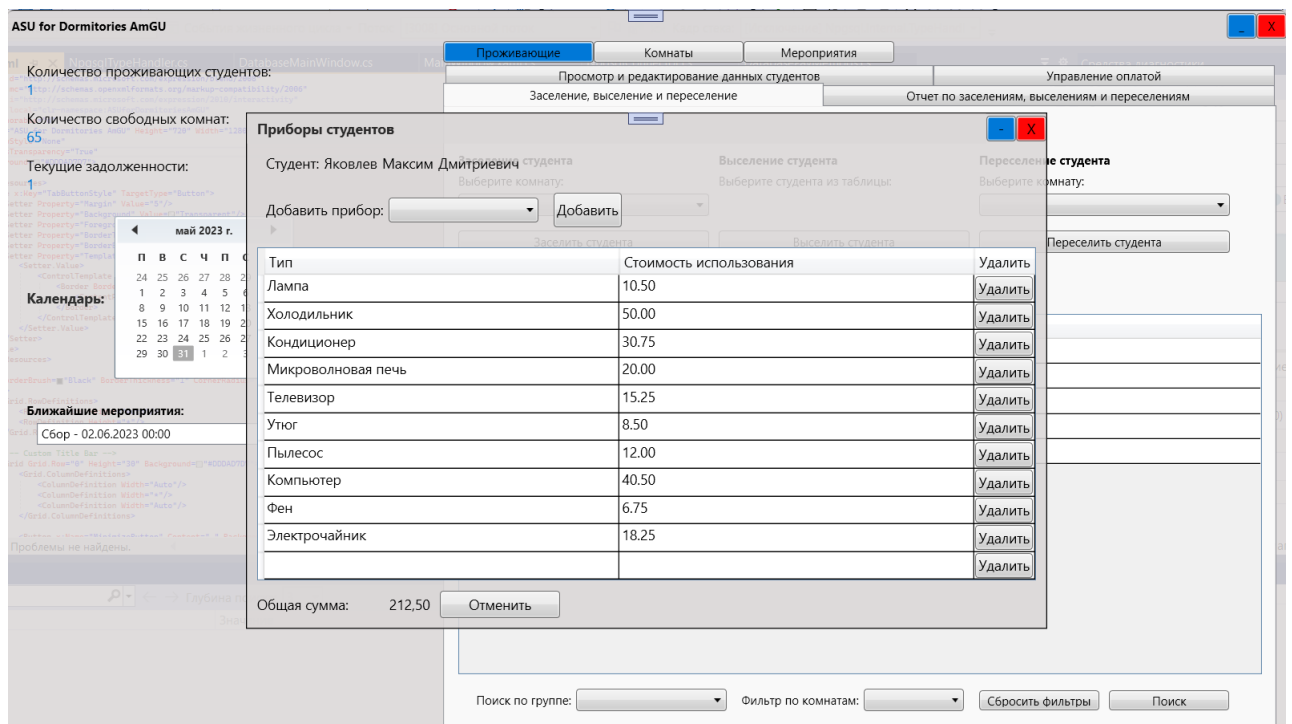


Рисунок В.8 – Список электроустановок выбранного студента

## Продолжение приложения В

The screenshot displays the ASU for Dormitories AmGU application. The interface is divided into several sections:

- Navigation:** Tabs for 'Проживающие', 'Комнаты', and 'Мероприятия'. The 'Проживающие' tab is active.
- Sub-sections:** 'Просмотр и редактирование данных студентов' and 'Управление оплатой'. The 'Отчет по заселениям, выселениям и переселениям' sub-section is selected.
- Summary:** Statistics for 'Количество проживающих студентов' (1), 'Количество свободных комнат' (65), and 'Текущие задолженности' (1).
- Calendar:** A calendar for May 2023 with a date picker set to 15.
- Upcoming Events:** 'Сбор - 02.06.2023 00:00'.
- Report Table:** A table with columns: Студент, Комната, Тип операции, Дата операции, Причина. It lists four operations:

Студент	Комната	Тип операции	Дата операции	Причина
Иван Петров Сергеевич	5/1	Заселение	5/31/2023 12:00:00 AM	
Елена Иванова Алексеевна	7/1	Заселение	5/31/2023 12:00:00 AM	
Алексей Смирнов Александрович	6/2	Переселение	5/31/2023 12:00:00 AM	
Алексей Смирнов Александрович	6/2	Заселение	5/31/2023 12:00:00 AM	
Яковлев Максим Дмитриевич	1/1	Заселение	5/30/2023 12:00:00 AM	

Рисунок В.9 – Раздел «Отчёт по заселениям, выселениям и переселениям»

# ПРИЛОЖЕНИЕ Г

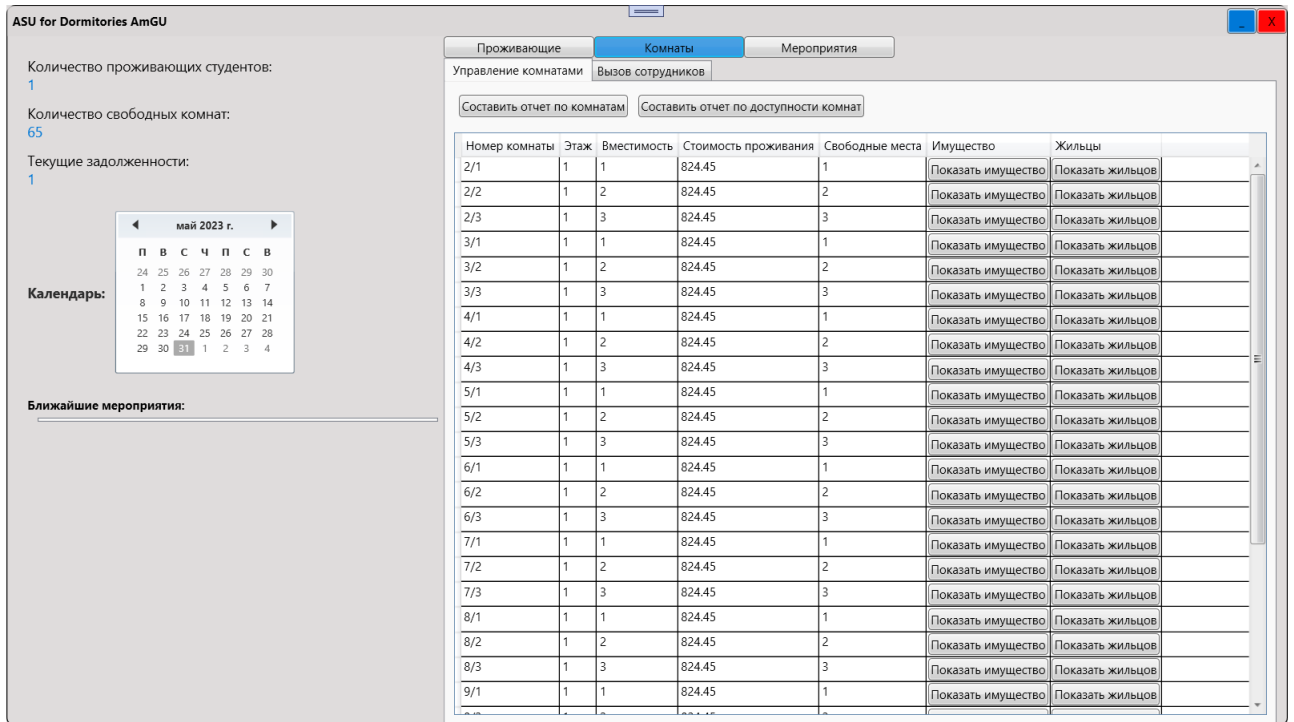


Рисунок Г.1 – Раздел «Комнаты»

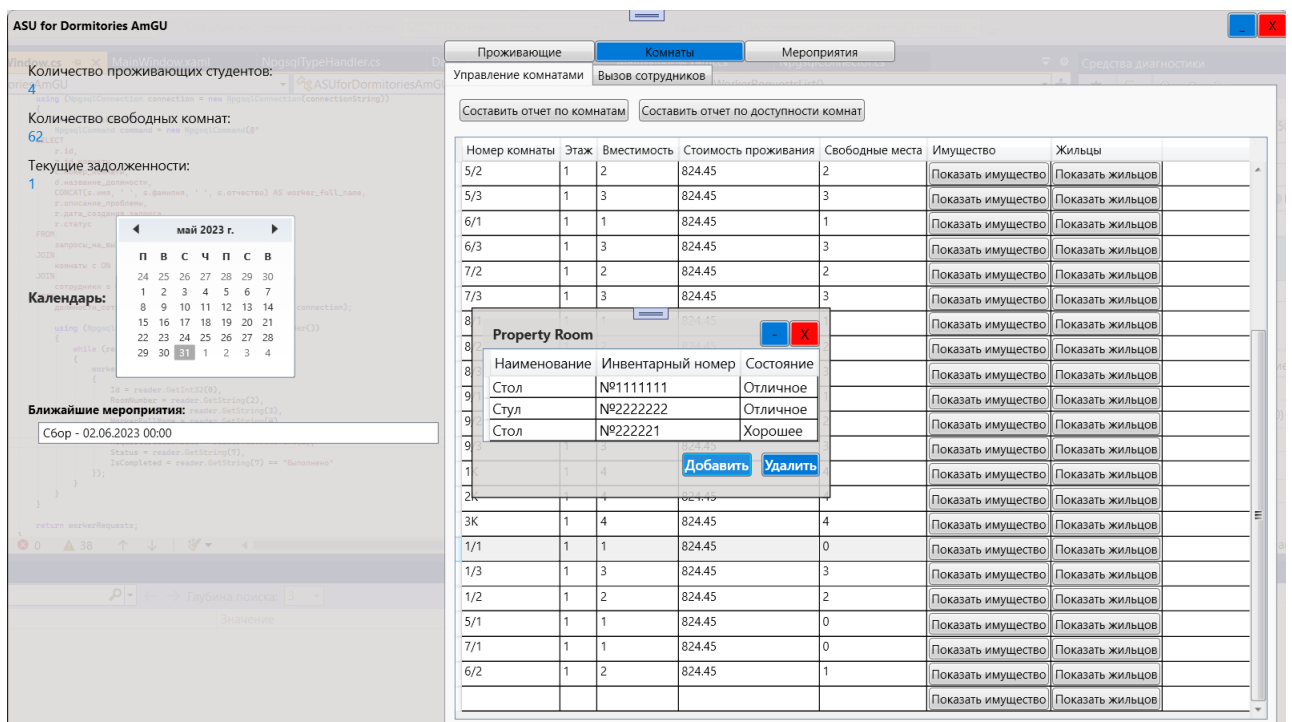


Рисунок Г.2 – Список имущества выбранной комнаты

## Продолжение приложения Г

ASU for Dormitories AmGU

Количество проживающих студентов: 1  
Количество свободных комнат: 65  
Текущие задолженности: 1

Календарь: май 2023 г.

Ближайшие мероприятия:

Проживающие | **Комнаты** | Мероприятия

Управление комнатами | Вызов сотрудников

Составить отчет по комнатам | Составить отчет по доступности комнат

Номер комнаты	Этаж	Вместимость	Стоимость проживания	Свободные места	Имущество	Жильцы
5/1	1	1	824.45	1	Показать имущество	Показать жильцов
5/2	1	2	824.45	2	Показать имущество	Показать жильцов
5/3	1	3	824.45	3	Показать имущество	Показать жильцов
6/1	1	1	824.45	1	Показать имущество	Показать жильцов
6/2	1	2	824.45	2	Показать имущество	Показать жильцов
6/3	1	3	824.45	3	Показать имущество	Показать жильцов
7/1	1	1	824.45	1	Показать имущество	Показать жильцов
8/2	1	2	824.45	2	Показать имущество	Показать жильцов
8/3	1	3	824.45	3	Показать имущество	Показать жильцов
9/1	1	1	824.45	1	Показать имущество	Показать жильцов
9/2	1	2	824.45	2	Показать имущество	Показать жильцов
9/3	1	3	824.45	3	Показать имущество	Показать жильцов
1K	1	4	824.45	4	Показать имущество	Показать жильцов
2K	1	4	824.45	4	Показать имущество	Показать жильцов
3K	1	4	824.45	4	Показать имущество	Показать жильцов
1/1	1	1	824.45	0	Показать имущество	Показать жильцов
1/3	1	3	824.45	3	Показать имущество	Показать жильцов
1/2	1	2	824.45	2	Показать имущество	Показать жильцов
					Показать имущество	Показать жильцов

**Таблица жильцов**

Имя	Фамилия	Отчество	Дата рождения	Номер телефона	Электронная почта	Группа
Яковлев	Максим	Дмитриевич	10/29/2001 12:00:0	+7 999 999 99 99	maksimka-yakovlev	953-о6

Рисунок Г.3 – Список жильцов выбранной комнаты

ASU for Dormitories AmGU

Количество проживающих студентов: 4  
Количество свободных комнат: 62  
Текущие задолженности: 1

Календарь: май 2023 г.

Ближайшие мероприятия: Сбор - 02.06.2023 00:00

Проживающие | **Комнаты** | Мероприятия

Управление комнатами | **Вызов сотрудников** | Средства диагностики

Создать запрос | Очистить журнал запросов | Открыть список сотрудников

Комната	Должность сотрудника	ФИО сотрудника	Описание проблемы	Дата создания зап
2/1	Сантехник	Кирилл Тягло Русланович	Неисправность в комнате 2/1. Течь крана в ванной комнате.	

Рисунок Г.4 – Раздел «Вызов сотрудников»

## Продолжение приложения Г

The screenshot shows the 'ASU for Dormitories AmGU' application interface. On the left, there are statistics: 'Количество проживающих студентов: 1', 'Количество свободных комнат: 65', and 'Текущие задолженности: 1'. Below this is a calendar for May 2023. The main area has tabs for 'Управление комнатами' and 'Вызов сотрудников'. The 'Вызов сотрудников' tab is active, showing a 'Запрос на ремонт' form. The form includes fields for 'Комната' (2/1), 'Сотрудник' (Кирилл Тягло Русланович (Сантехник)), and 'Описание проблемы' (Неисправность в комнате 2/1. Течь крана в ванной комнате). A 'Создать запрос' button is at the bottom.

Рисунок Г.5 – Запрос на вызов сотрудника

This screenshot shows the same application interface as Figure G.5, but with a confirmation dialog box overlaid. The dialog box contains the text: 'Запрос на вызов сотрудника успешно создан и выслан на электронную почту сотрудника.' and an 'OK' button. The background form is dimmed.

Рисунок Г.6 – Отправка письма сотруднику о проблеме

## Продолжение приложения Г

### Уведомление о запросе на ремонт

-  Общежитие 2 ФГБОУ ВО АмГУ Сегодня, 18:16 Кому: вам 🔖 🗨 📅 📄 ...

Уважаемый Кирилл Тягло Русланович,

Вам поступил запрос на ремонт в комнате 2/1. Описание проблемы: Неисправность в комнате 2/1. Течь крана в ванной комнате.

С уважением,  
Администрация общежития

### Рисунок Г.7 – Пришедшее письмо сотруднику с проблемой

## ПРИЛОЖЕНИЕ Д

ASU for Dormitories AmGU

Количество проживающих студентов: 1

Количество свободных комнат: 65

Текущие задолженности: 1

Календарь: май 2023 г.

п	в	с	ч	п	с	в
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Ближайшие мероприятия:

Проживающие | Комнаты | **Мероприятия**

Планирование мероприятий (Заполнять после настройки оповещений)

Название мероприятия:

Сбор

Описание мероприятия:

Сбор в холле на 2 этаже. По поводу субботника.

Дата и время проведения: 15.04.2023

Место проведения:

Прилегающая территория к общежитию №2

Добавить мероприятие

Оповещения о мероприятиях (Заполнять до создания мероприятия)

Тип оповещения:

Отправить на почту  
maksimka-yakovlev-2001@mail.ru

Запомнить адрес электронной почты

Отчет по мероприятиям

Выберите период отчета:

Выбор даты: 15

до

Выбор даты: 15

Создать отчет

Рисунок Д.1 – Вкладка «Мероприятия»

ASU for Dormitories AmGU

Количество проживающих студентов: 1

Количество свободных комнат: 65

Текущие задолженности: 1

Календарь: май 2023 г.

п	в	с	ч	п	с	в
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Ближайшие мероприятия:

Сбор - 02.06.2023 00:00

Рисунок Г.2 – Появившееся мероприятие в панели состояния после создания