

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«**СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ**»

Хакасский технический институт – филиал ФГАОУ ВО  
«Сибирский федеральный университет»

Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ О. В. Папина  
подпись  
«\_\_\_\_\_» \_\_\_\_\_ 2023 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.03 Прикладная информатика

Разработка информационной системы планирования событий  
Министерства здравоохранения Республики Хакасия

Руководитель \_\_\_\_\_  
подпись, дата

ст. преподаватель В. И. Кокова

Выпускник \_\_\_\_\_  
подпись, дата

С. И. Боев

Консультанты  
по разделам:

Экономический \_\_\_\_\_  
подпись, дата

М. А. Бурева

Нормоконтролер \_\_\_\_\_  
подпись, дата

А. Н. Кадычегова

Абакан 2023

Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Хакасский технический институт – филиал ФГАОУ ВО  
«Сибирский федеральный университет»

Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ О. В. Папина  
подпись  
« \_\_\_\_ » \_\_\_\_\_ 2023 г.

**ЗАДАНИЕ  
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ  
в форме бакалаврской работы**

Абакан 2023

Студенту Боеву Сергею Ивановичу

Группа ХБ 19-02

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка информационной системы планирования событий Министерства здравоохранения Республики Хакасия

Утверждена приказом по институту № 283 от 11.05.2023 г.

Руководитель ВКР: В. И. Кокова, ст. преподаватель, ХТИ – филиал СФУ

Исходные данные для ВКР: заказ ГБУЗ РХ «РМИАЦ».

Перечень разделов ВКР:

1. Анализ предметной области. Выбор средств проектных решений.
2. Описание разработки информационной системы планирования событий Министерства здравоохранения республики Хакасия.
3. Определение совокупной стоимости владения информационной системой планирования событий Министерства здравоохранения Республики Хакасия.

Перечень графического материала: нет

Руководитель ВКР

\_\_\_\_\_

подпись

В. И. Кокова

Задание принял к исполнению

\_\_\_\_\_

подпись

С. И. Боев

«11» мая 2023г.

## РЕФЕРАТ

Выпускная квалификационная работа по теме: «Разработка информационной системы планирования событий Министерства здравоохранения Республики Хакасия» содержит 80 страниц текстового документа, 11 использованных источников, 56 иллюстраций, 7 таблиц, 8 формул.

IDEF3, DFD, ЧАТ-БОТ, ИНФОРМАЦИОННАЯ СИСТЕМА, ТАБЛИЦЫ, МИНЗДРАВ, РМИАЦ, C#, POSTGRESQL, VISUAL STUDIO, БАЗА ДАННЫХ, КАПИТАЛЬНЫЕ ЗАТРАТЫ, ТСО, РИСКИ, СОБЫТИЯ.

Объект выпускной квалификационной работы (ВКР): деятельность Минздрава РХ и ГБУЗ РХ «РМИАЦ» по планированию внутриорганизационных событий.

Предмет: веб-приложение и чат-бот «Информационная система планирования событий».

Цель выпускной квалификационной работы: разработка информационной системы планирования событий Министерства здравоохранения Республики Хакасия для ускорения и упрощения данного процесса.

Задачи:

1. Провести анализ предметной области. Выбрать средства проектных решений.
2. Разработать информационную систему «Планировщик событий», а именно: веб-приложение и чат-бот Telegram.
3. Описать разработку информационной системы.
4. Определить совокупную стоимость владения системой.

В итоге были разработаны база данных, веб-приложение и чат-бот Telegram, которые в своей совокупности образуют информационную систему планирования событий Министерства здравоохранения Республики Хакасия. Данная система значительно оптимизирует процесс планирования событий.

## SUMMARY

The graduation thesis is «Event-scheduling IT System for Ministry of Health of Republic of Khakassia». It contains 80 pages, 11 references, 56 figures, 7 tables, 8 formulae.

IDEF3, DFD, CHAT-BOT, INFORMATION SYSTEM, TABLES, MINISTRY OF HEALTH, RMIAC, C#, POSTGRESQL, VISUAL STUDIO, DATABASE, CAPITAL EXPENDITURES, TCO, RISKS, EVENTS

The object of the thesis: the work activity of the Ministry of Health of the Republic of Khakassia and the State Medical Institution of the Republic of Khakassia on the planning of intra-organizational events.

The purpose of the graduation work is to develop a web application and a chat-bot “Event-scheduling IT System”.

The purpose of the graduation work is to develop an event-scheduling IT system for of the Ministry of Health of the Republic of Khakassia to speed up and simplify this process.

Objectives:

1. To conduct an analysis of the subject area; to choose the means of design solutions.
2. To develop an IT system “Event Planner”, namely: a web application and a Telegram chat-bot.
3. To describe the development of the IT system.
4. To determine the total cost of ownership of the IT system.

Results: a database, a web application and a Telegram chat-bot have been developed; they form the event planning IT system of the Ministry of Health of the Republic of Khakassia. This system significantly optimizes the event planning process.

English language supervisor

N. V. Chezybaeva

---

подпись, дата

## СОДЕРЖАНИЕ

Введение.....	6
1 Анализ предметной области. Выбор средств проектных решений .....	7
1.1 Анализ основных процессов Республиканского медицинского информационно-аналитического центра Республики Хакасия и Министерства здравоохранения Республики Хакасия .....	7
1.1.1 О Министерстве здравоохранения Республики Хакасия.....	7
1.1.2 Характеристика деятельности ГБУЗ РХ «РМИАЦ» .....	9
1.2Актуальность разработки. Разработка концепции IT-проекта.....	14
1.3Анализ аналогичных информационных систем и обоснование необходимости собственной разработки.....	15
1.4 Моделирование для описания бизнес-процесса планирования событий..	18
1.4.1 Моделирование действий пользователя в нотации IDEF3 .....	18
1.4.2Моделирование потоков данных в нотации DFD.....	23
1.5Описание информационно-логической модели данных ER.....	28
1.6 Анализ средств разработки информационных систем.....	29
1.6.1 Выбор архитектуры.....	29
1.6.2 Выбор языка программирования .....	30
1.6.3 Выбор системы управления базами данных (СУБД) .....	32
1.7 Выводы по разделу: «Анализ предметной области. Выбор средств проектных решений» .....	35
2 Описание разработки информационной системы планирования событий Министерства здравоохранения Республики Хакасия.....	36
2.1 Описание разработки базы данных для информационной системы планирования событий Министерства здравоохранения Республики Хакасия	36

2.2	Описание разработки чат-бота Telegram для информационной системы планирования событий Министерства здравоохранения Республики Хакасия	41
2.3	Описание разработки веб-приложения для информационной системы планирования событий Министерства здравоохранения Республики Хакасия	55
2.4	Выводы по разделу «Описание разработки информационной системы планирования событий Министерства здравоохранения Республики Хакасия»	63
3	Определение совокупной стоимости владения информационной системой планирования событий Министерства здравоохранения Республики Хакасия	63
3.1	Расчет затрат на реализацию проекта	64
3.2	Капитальные затраты	65
3.3	Расчет эксплуатационных затрат	71
3.4	Прямые затраты	72
3.5	Выводы по разделу «Определение совокупной стоимости владения информационной системой планирования событий Министерства здравоохранения Республики Хакасия»	74
	Заключение	75
	Список использованных источников	76

## ВВЕДЕНИЕ

На рынке существует множество систем для автоматизации планирования работ компании, но зачастую эти решения сложно встроить в уже существующую инфраструктуру предприятия. В них не проработаны процессы взаимодействия участников проекта. А в случае с использованием в корпоративных сетях организаций разнородных программ и систем, внедрение специализированного решения неудобно. Из-за таких недостатков организации начинают искать различные пути решения такой проблемы, например, разработка собственной информационной системы (ИС) планирования событий. Преимущество разработки собственной ИС заключается в том, что организация может создать такую информационную систему, которая лучше любой другой подходила бы под ее основные внутренние процессы. Такая система позволит уменьшить затраты на автоматизацию данного бизнес-процесса, а также оптимизировать и упростить его.

Объект выпускной квалификационной работы: деятельность Министерства здравоохранения Республики Хакасия и Республиканского медицинского информационно-аналитического центра Республики Хакасия по планированию внутриорганизационных событий.

Цель выпускной квалификационной работы: разработка информационной системы планирования событий Министерства здравоохранения Республики Хакасия для ускорения и упрощения данного процесса.

Для достижения цели необходимо выполнить следующие задачи:

1. Провести анализ предметной области. Выбрать средства проектных решений.
2. Разработать информационную систему «Планировщик событий», а именно: веб-приложение и чат-бот Telegram.
3. Описать разработку информационной системы.
4. Определить совокупную стоимость владения информационной системой.



## **1 Анализ предметной области. Выбор средств проектных решений**

Государственное бюджетное учреждение здравоохранения Республики Хакасия «Республиканский медицинский информационно-аналитический центр» (ГБУЗРХ «РМИАЦ») - образован в 2000 году для предоставления услуг по обработке данных, размещению информации и связанной с этим, деятельностью.

Учредитель ГБУЗ РХ «РМИАЦ» – Министерство здравоохранения Республики Хакасия. Для определения бизнес-процессов, а также выявления среди них того, который можно было бы автоматизировать, необходимо изучить основную деятельность Министерства здравоохранения в области автоматизации, а также определить роль ГБУЗ РХ «РМИАЦ» в этом процессе. После этого следует выбрать программные средства для разработки АИС и определить архитектуру АИС.

### **1.1 Анализ основных процессов Республиканского медицинского информационно-аналитического центра Республики Хакасия и Министерства здравоохранения Республики Хакасия**

#### **1.1.1 О Министерстве здравоохранения Республики Хакасия**

Министерство здравоохранения Республики Хакасия является исполнительным органом государственной власти Республики Хакасия, функции которого относятся к реализации государственной политики и управлению в сфере здравоохранения. Они включают вопросы организации оказания медицинской помощи и лекарственного обеспечения, на основании Положения «О Министерстве здравоохранения Республики Хакасия», утвержденном Постановлением Правительства Республики Хакасия от 11 июня 2009 г. № 260. Основными задачами Министерства здравоохранения Республики Хакасия являются:

- обеспечение реализации конституционных прав граждан Российской Федерации на гарантированное получение медицинской помощи, лекарственных средств, изделий медицинского назначения;
- реализация государственной политики и государственного управления в области здравоохранения;
- обеспечение населения своевременной, доступной, качественной медицинской помощью [2].

Структурная схема Министерства здравоохранения Республики Хакасия представлена на рисунке 1.

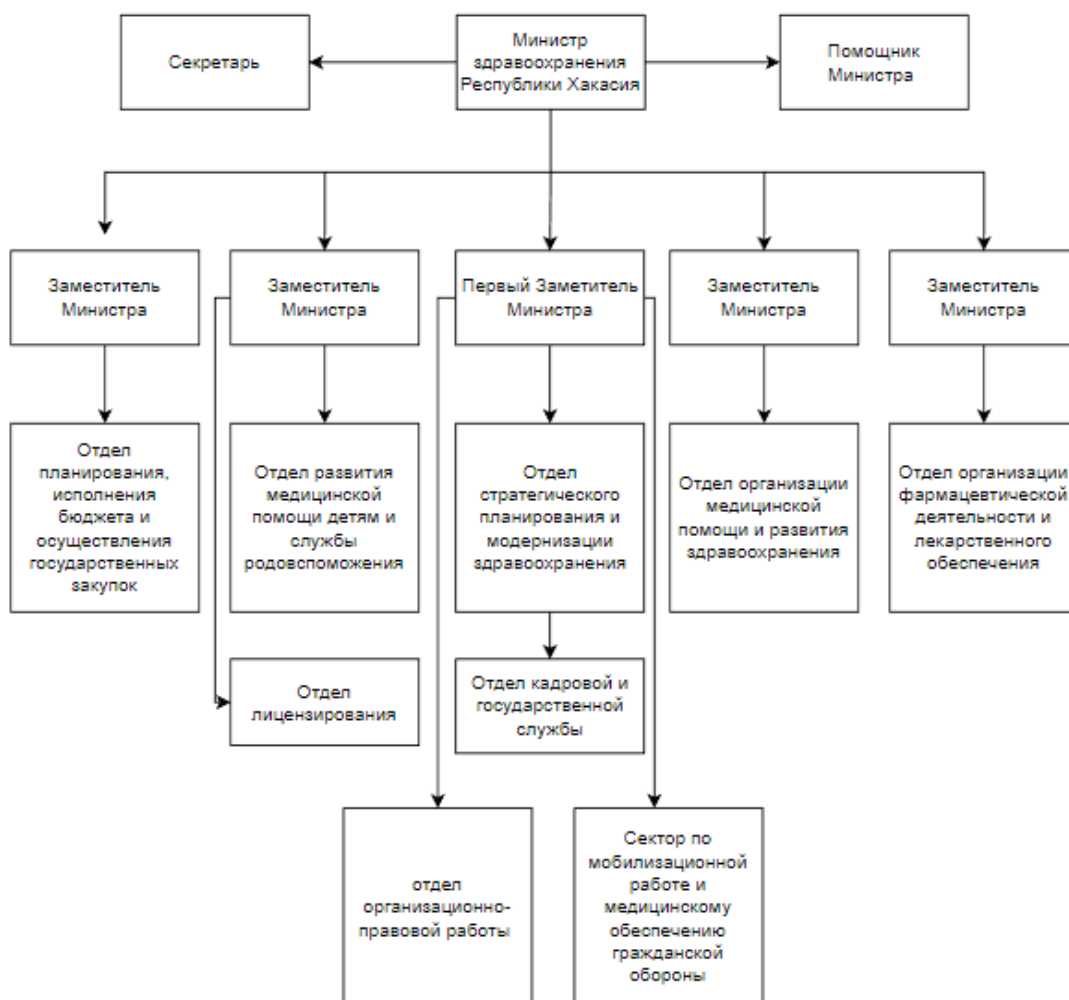


Рисунок 1 – Структурная схема Министерства здравоохранения Республики Хакасия

Министерство осуществляет свою деятельность во взаимодействии с федеральными органами исполнительной власти, в том числе территориальным органом Федеральной службы по надзору в сфере здравоохранения и социального развития, территориальным органом Федеральной службы по надзору в сфере защиты прав потребителей и благополучия человека, органами исполнительной власти Республики Хакасия, органами местного самоуправления, Территориальным фондом обязательного медицинского страхования Республики Хакасия, Государственным учреждением – Отделением Пенсионного фонда Российской Федерации по Республике Хакасия, Государственным учреждением – Региональным отделением Фонда социального страхования Российской Федерации по Республике Хакасия, медицинскими организациями, предприятиями, учреждениями, организациями, общественными объединениями и иными некоммерческими организациями, осуществляющими свою деятельность в сфере охраны здоровья [3].

### **1.1.2 Характеристика деятельности ГБУЗ РХ «РМИАЦ»**

Государственное бюджетное учреждение здравоохранения Республики Хакасия «Медицинский информационно-аналитический центр» создано для выполнения функций по обеспечению реализации предусмотренных законодательством Российской Федерации полномочий в сфере организации сбора, обработки и анализа медицинских статистических данных о деятельности учреждений здравоохранения, состоянии здоровья населения, управления системой медицинского статистического учета и отчетности в организациях и учреждениях здравоохранения Республики Хакасия.

ГБУЗ РХ «РМИАЦ» является некоммерческой организацией, созданной в соответствии с Гражданским кодексом Российской Федерации, Федеральным законом от 12.01.1996 № 7-ФЗ «О некоммерческих организациях», и не преследует извлечение прибыли в качестве основной цели своей деятельности.

Предметом деятельности Учреждения является организация сбора, обработки и анализа медицинских статистических данных о деятельности учреждений здравоохранения, состоянии здоровья населения, управление системой медицинского статистического учета и отчетности в организациях и учреждениях здравоохранения Республики Хакасия.

Основными целями деятельности ГБУЗ РХ «РМИАЦ» являются:

– организационно-методическое руководство по формированию единой информационной системы здравоохранения в Республике Хакасия, созданию и сопровождению автоматизированных систем управления здравоохранением Республики Хакасия;

– координация деятельности службы медицинской статистики и медицинского статистического обеспечения органов управления здравоохранением Республики Хакасия;

– формирование и сопровождение государственной и отраслевой статистической отчетности Российской Федерации;

– анализ медико-статистической информации о состоянии здоровья населения и деятельности здравоохранения в Республике Хакасия;

– повышение эффективности использования информационной структуры здравоохранения Республики Хакасия, обеспечение достоверности сведений в учетной и отчетной медицинской документации;

– внедрение новых технологий сбора и обработки медико-статистической информации;

– создание и поддержание единого набора классификаторов системы мониторинга здоровья населения;

– обеспечение безопасности автоматизированных информационных систем здравоохранения Республики Хакасия;

– обеспечение информационной поддержки по реализации приоритетного национального проекта «Здоровье»;

– участие в подготовке ежегодного доклада о состоянии общественного здоровья граждан Российской Федерации, проживающих на территории

Республики Хакасия, и представление его в правительство Республики Хакасия;

– представление в Министерство здравоохранения и социального развития Российской Федерации сводных государственных и отраслевых медицинских статистических отчетов от Республики Хакасия в соответствии с установленным порядком;

– формирование единой информационной системы здравоохранения Российской Федерации на базе современных компьютерных технологий межотраслевой системы: сбора, обработки, хранения и предоставления информации, обеспечивающей динамическую оценку состояния здоровья и информационную поддержку принятия решений, направленных на его улучшение.

Для достижения данных целей ГБУЗ РХ «РМИАЦ» осуществляет следующие виды деятельности:

– разработка, планирование, координация в области информатизации здравоохранения, медицинской статистики;

– разработка и реализация концепций и программ информатизации здравоохранения Республики Хакасия;

– координация работ по созданию единой информационной базы системы здравоохранения и обязательного медицинского страхования Республики Хакасия;

– формирование единой системы учета и отчетности медико-статистической информации с применением новых технологий сбора и обработки информации;

– прием, обработка статистических отчетов от органов управления и учреждений здравоохранения Республики Хакасия;

– разработка показателей, характеризующих деятельность учреждений здравоохранения Республики Хакасия, состояние здоровья населения и здравоохранения в рамках утвержденной статистической отчетности;

– разработка, внедрение и сопровождение автоматизированных систем

сбора, обработки, хранения и передачи информации;

–анализ полученной информации с привлечением главных специалистов органов управления здравоохранением Республики Хакасия, организационно-методических отделов республиканских и муниципальных лечебно-профилактических учреждений здравоохранения;

–осуществление взаимодействия с территориальным фондом обязательного медицинского страхования, территориальными органами государственной статистики, образовательными и научными учреждениями, учреждениями госсанэпидслужбы и другими сторонними организациями.

–контроль за использованием в работе учреждений здравоохранения;

–международных классификаций при ведении медицинской документации;

–контроль за ведением в учреждениях здравоохранения медицинской документации, удостоверяющей случаи смерти;

–обеспечение подведомственных учреждений здравоохранения установленными формами медицинской учетной и отчетной документации;

–изучение и прогнозирование процессов и явлений, связанных со здоровьем человека;

–выработка предложений по повышению эффективности деятельности учреждений здравоохранения;

–организация и контроль за состоянием статистического учета и отчетности учреждений здравоохранения Республики Хакасия;

–подготовка и публикация аналитических обзоров о состоянии здоровья и здравоохранения, сборников основных показателей по разделам здравоохранения;

–проведение мероприятий по повышению деловой и профессиональной квалификации работников службы медицинской статистики и технического обеспечения всех уровней, организация семинаров, распространение передового опыта и оказание постоянной организационно-методической помощи;

–участие в разработке стандартов медицинской помощи, базовой программы обязательного медицинского страхования, а также целевых программ охраны здоровья населения;

–создание и ведение баз данных нормативных и распорядительных документов по здравоохранению, выполнение работ по тиражированию документации;

–обучение работников системы здравоохранения Республики Хакасия информационным технологиям;

–организация работ по построению единой системы телекоммуникаций, техническому и информационному обеспечению мероприятий, проводимых в системе здравоохранения Республики Хакасия;

–приобретение и обслуживание компьютерного и периферийного оборудования [4].

В данном разделе проанализирована основная деятельность Минздрава Хакасии и его подведомственного учреждения ГБУЗ РХ «Республиканский медицинский информационно-аналитический центр». Из описания процессов можно сделать вывод, что в организации достаточно много различных обязанностей.

Большое количество обязанностей предполагает их распределение по сотрудникам. Распределение происходит на различных заседаниях, совещаниях и т.п. Обычно данные совещания назначаются лично, и таких совещаний может быть достаточно много и их приходится все запоминать, что достаточно неудобно, так как при различных обстоятельствах, что-то может забыться. Исходя из вышесказанного, можно сделать вывод о том, что в качестве бизнес-процесса, подлежащего автоматизации можно выбрать процесс планирования событий, таких как заседания, совещания, собрания, мероприятия, конференции и т.п.

## 1.2 Актуальность разработки. Разработка концепции IT-проекта

Актуальность автоматизации процесса «Планирование событий» обосновывается тем, что на данный момент организация не располагает такой ИС, которая решала бы большую часть основных внутренних проблем планирования событий. К таким проблемам можно отнести:

- сотрудникам помимо своей работы приходится помнить или записывать о различных собраниях, мероприятиях, конференциях и т.п., что в свою очередь может отвлекать от работы и доставлять неудобство;

- человеку, который оповещает сотрудников о предстоящем событии, приходится каждому лично доносить информацию, что занимает достаточно большое количество времени;

- в случае, если сотрудник вдруг забыл о событии, вероятность того, что ему напомнят, достаточно мала;

- в случае, если событие не состоится, приходится также оповещать каждого сотрудника лично.

Идея заключается в том, чтобы создать такую информационную систему, которая выполняла бы следующие функции:

- оповещение участников о новом событии посредством мессенджера Telegram;

- создание, редактирование и удаление событий одним человеком;

- возможность задавать дату, время и место встречи;

- просмотр предстоящих событий;

- просмотр истории событий;

- добавление новых пользователей ИС.

Такая информационная система поможет решить проблемы, описанные ранее, а также ускорить и упростить процесс планирования событий.



### 1.3 Анализ аналогичных информационных систем и обоснование необходимости собственной разработки

В качестве информационных систем аналогов для анализа были выбраны следующие информационные системы: стандартный мобильный календарь и Google календарь. На рисунках 2 и 3 представлены скриншоты данных программ.

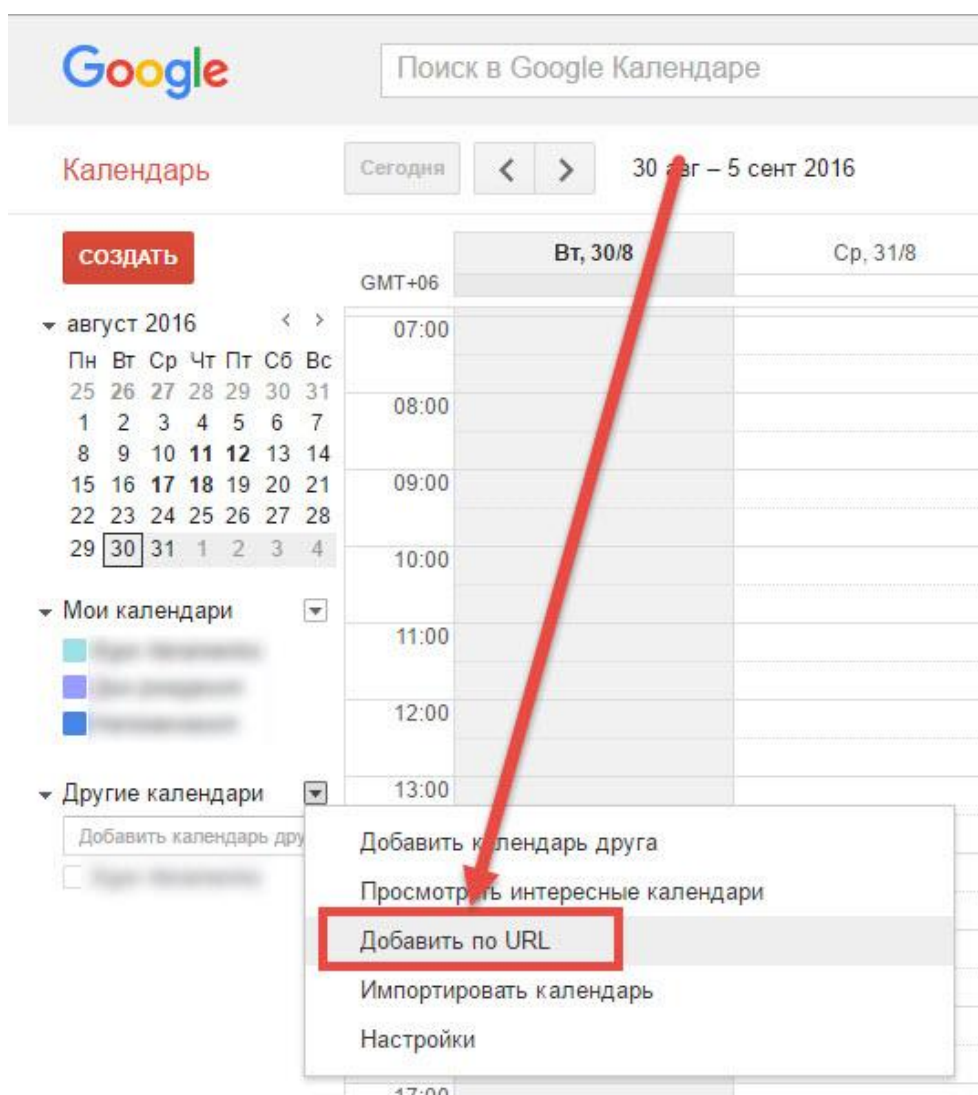


Рисунок 2 – Google календарь

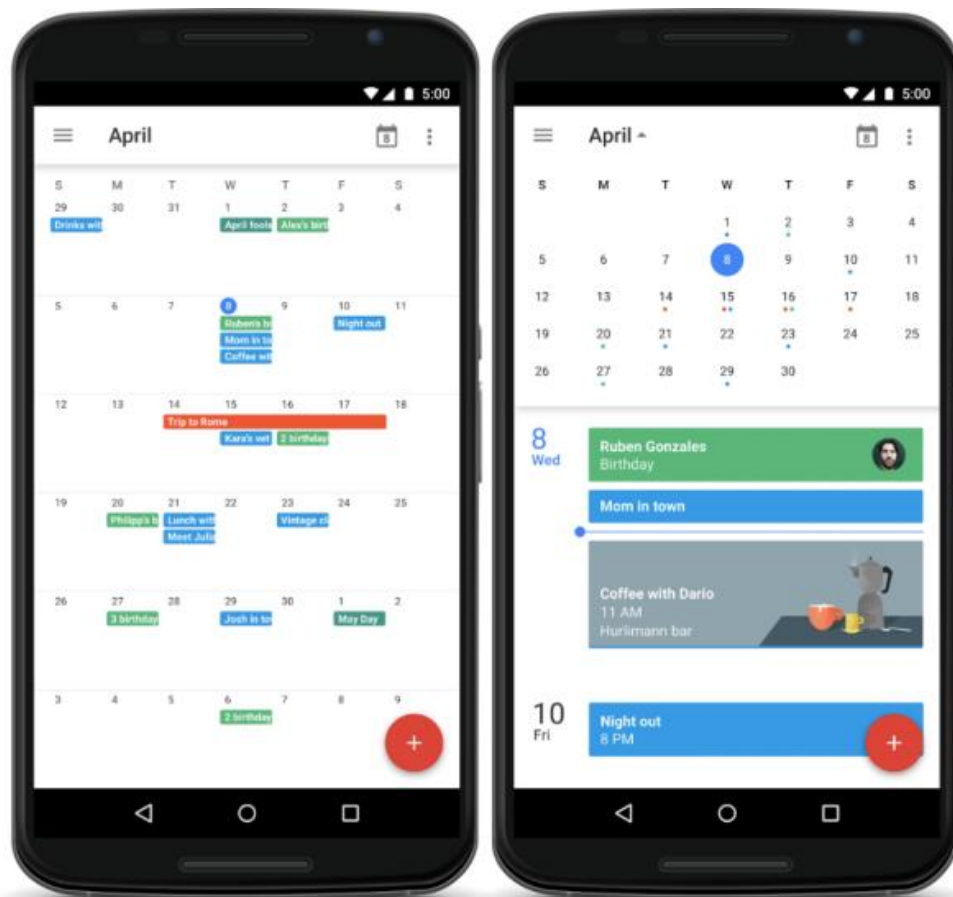


Рисунок 3 – Стандартный мобильный календарь

Google календарь – сервис для планирования встреч, событий и дел, разработанный компанией Google.

Рассмотрим основные функции данной информационной системы.

Основные функции Google календарь:

- оповещение о предстоящих событиях;
- возможность создания, редактирования и удаления события;
- возможность просмотра событий на различных промежутках времени;
- возможность задавать время и место встречи;
- возможность приглашать других участников;
- просмотр истории событий;
- оповещение участников событий по электронной почте.

Недостатком информационной системы Google календарь является то, что вся информация о пользователях и встречах хранится на серверах Google, а не Министерства здравоохранения РХ. Также к недостаткам можно отнести отсутствие возможности оповещения посредством мессенджера Telegram, так как это является одной из необходимых функций, которые должна выполнять требуемая ИС. Данная информационная система обладает также и другими функциями, но из необходимых для сравнения функций присутствуют только вышеперечисленные.

Календарь – мобильное приложение календаря с разделением на полугодия, кварталы, месяцы и недели с функцией производственного календаря. В нем отмечены праздники, короткие дни, рабочие дни и дни отпуска.

В календарь можно добавлять события в виде заметок и настраивать для них время показа уведомлений.

Рассмотрим основные функции информационной системы «Календарь». Основные функции мобильного календаря:

- возможность создания, редактирования и удаления события;
- возможность просмотра событий на различных промежутках времени;
- возможность задавать время и место встречи;
- оповещение о предстоящих событиях;
- просмотр истории событий.

К недостаткам данной ИС можно отнести:

–Отсутствие возможности назначения участников событий, от чего появляется следующий недостаток – отсутствие возможности оповещения участников события.

–Также данная ИС находится на мобильном устройстве и никак не взаимодействует с сетью интернет, что говорит о том, что если вдруг сотрудник оставил свой телефон дома, то у него не будет возможности просмотра предстоящих событий.

–Как и предыдущая система, мобильный календарь не решает все

основные проблемы по планированию событий, а значит, возникает потребность в разработке собственной информационной системы.

## 1.4 Моделирование для описания бизнес-процесса планирования событий

### 1.4.1 Моделирование действий пользователя в нотации IDEF3

Модель действий IDEF3 отображает информационные потоки, а также взаимоотношение между процессами обработки информации и объектами, иначе говоря, это модель действий пользователя в ИС. В данной системе есть три уровня доступа: администратор, модератор, пользователь. Для того чтобы рассмотреть все функции системы, система будет рассматриваться с точки зрения администратора, так как данный уровень доступа имеет право пользоваться всеми функциями системы.

На рисунке 4 представлена диаграмма IDEF3 веб-приложения.

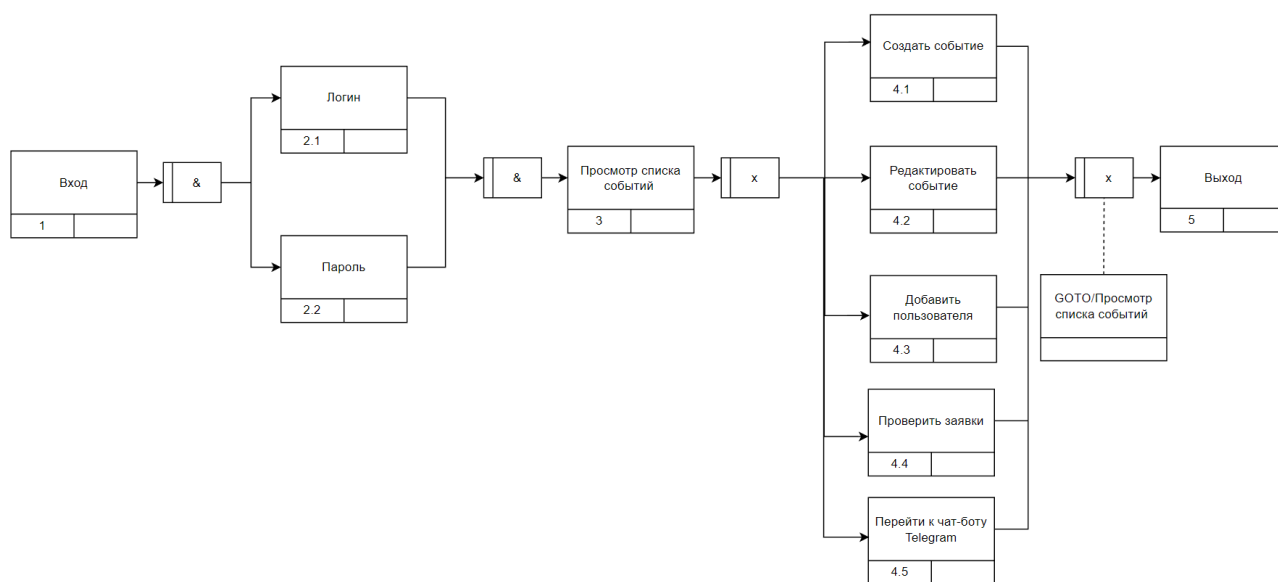


Рисунок 4 – Диаграмма IDEF3 веб-приложения

Пользователь открывает ИС «Планировщик событий», система предлагает ему авторизоваться, для того, чтобы определить права пользователя

и предоставить доступ к разрешенным ему функциям. После того как пользователь авторизовался, открывается окно просмотра списка событий и шесть возможных вариантов действий, а именно: выйти из своего аккаунта, создать событие, редактировать событие, добавить пользователя, проверить заявки, поступившие с чат-бота Telegram, перейти к чат-боту Telegram. В зависимости от прав доступа возможности пользователя будут изменяться.

С уровнем доступа «Пользователь» доступны только «Просмотр событий», «Перейти к чат-боту Telegram» и «Выход из аккаунта». Уровень доступа «Модератор» помимо функций «Пользователя» открывает доступ к функциям создания и редактирования события.

Уровень доступа «Администратор» имеет доступ ко всем функциям, доступным «Модератору», а также открывается функция «Добавить пользователя».

Функции «Создать событие», «Редактировать событие», «Удалить событие» и «Просмотр списка событий». При выборе функции «Создать событие» пользователю необходимо ввести данные, показанные на рисунке 5 в блоках 4.1.1.1-4.1.1.8. Когда все данные введены, необходимо назначить участников события. После выбора участников события, пользователь нажимает на кнопки «Добавить участников» и «Создать событие», тем временем система переходит на страницу просмотра событий и в данном окне появляется событие, созданное пользователем.

Система предоставляет возможность просмотра событий в нескольких представлениях:

- используя календарь для навигации;
- список всех событий.

Также у пользователя ИС, с соответствующим уровнем доступа, есть возможность редактировать события. Редактировать событие можно найдя нужное событие в окне просмотра событий, нажав кнопку «Редактировать событие».

Редактирование события происходит аналогично созданию события,

только все окна предварительно заполнены информацией о событии, которое было выбрано для редактирования.

Удалить событие можно аналогично редактированию, только вместо кнопки «Редактировать событие», необходимо нажать кнопку «Удалить событие».

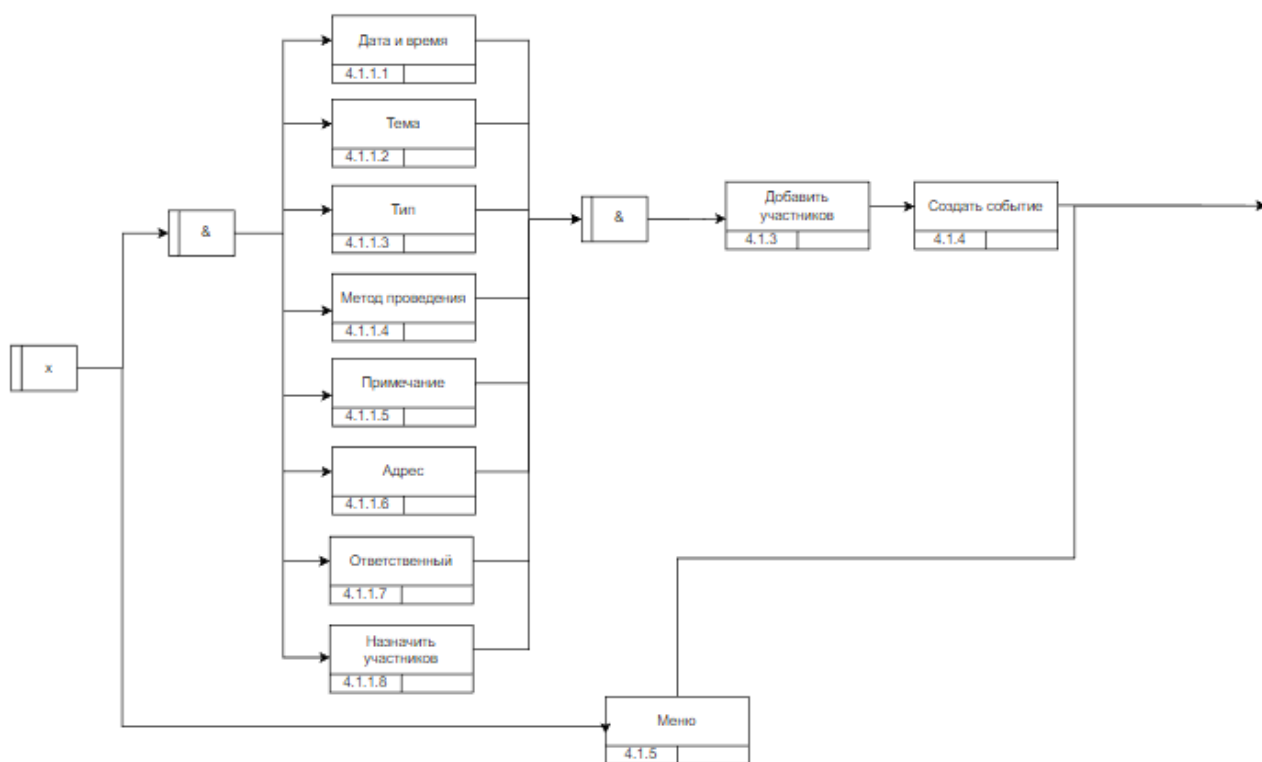


Рисунок 5 – Декомпозиция блока «Создать событие»

Функция «Добавить пользователя». При выборе данной функции пользователю необходимо заполнить поля, показанные на рисунке 6 в блоках 4.3.1.1 – 4.3.1.10, затем нажать кнопку «Создать учетную запись» и пользователь будет добавлен.

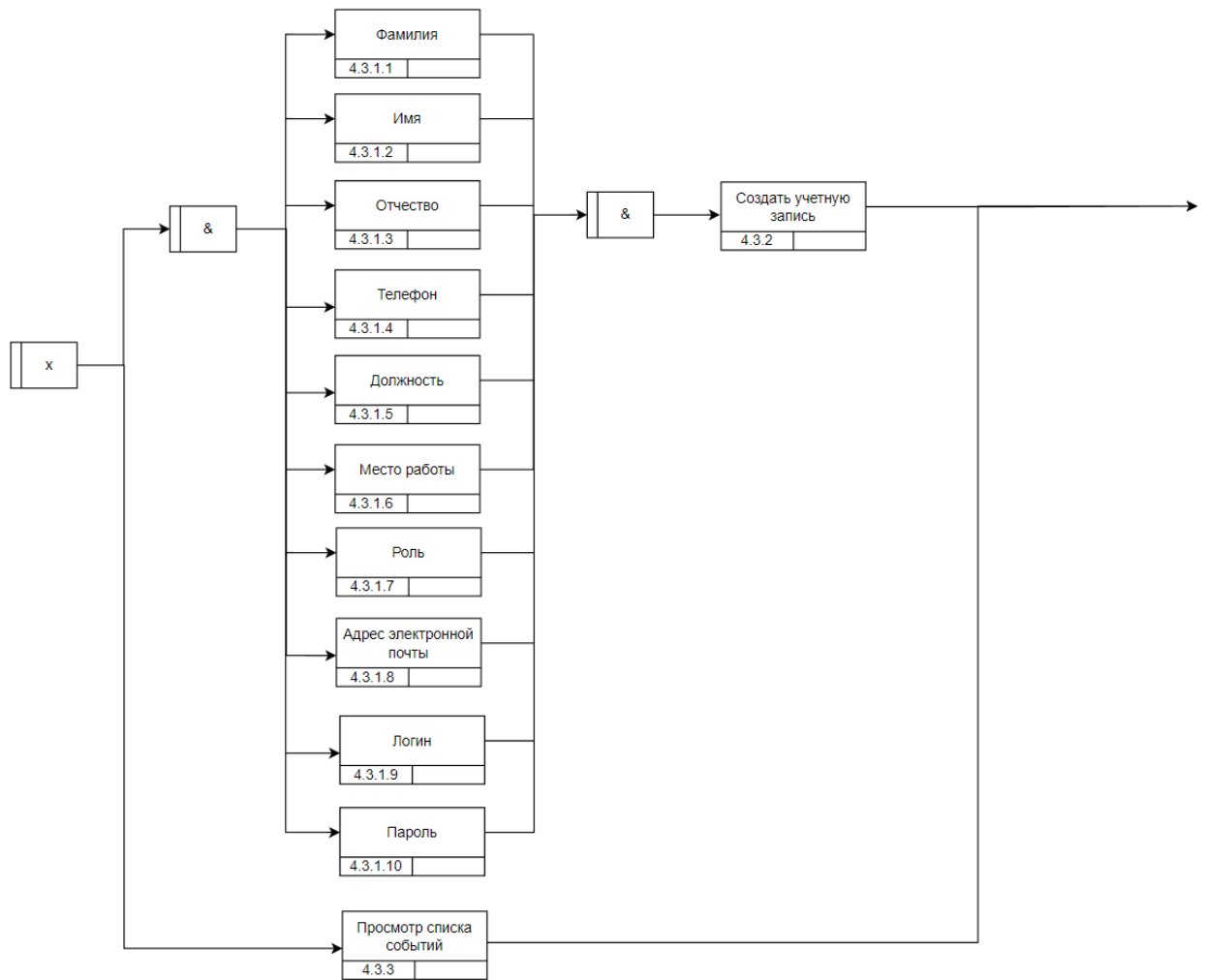


Рисунок 6 – Деконпозиция блока «Добавить пользователя»

На рисунке 7 представлена диаграмма IDEF3 чат-бота.



Рисунок 7 – Диаграмма IDEF3 чат-бота

Когда пользователь открывает чат-бота, он может либо авторизоваться, если был зарегистрирован в системе через веб-приложения, либо зарегистрироваться. Авторизация необходима для получения оповещений о событиях через чат-бота. Если же пользователь регистрируется через Telegram чат-бота самостоятельно, то авторизоваться ему уже нет необходимости.

На рисунке 8 представлена декомпозиция блока регистрации.

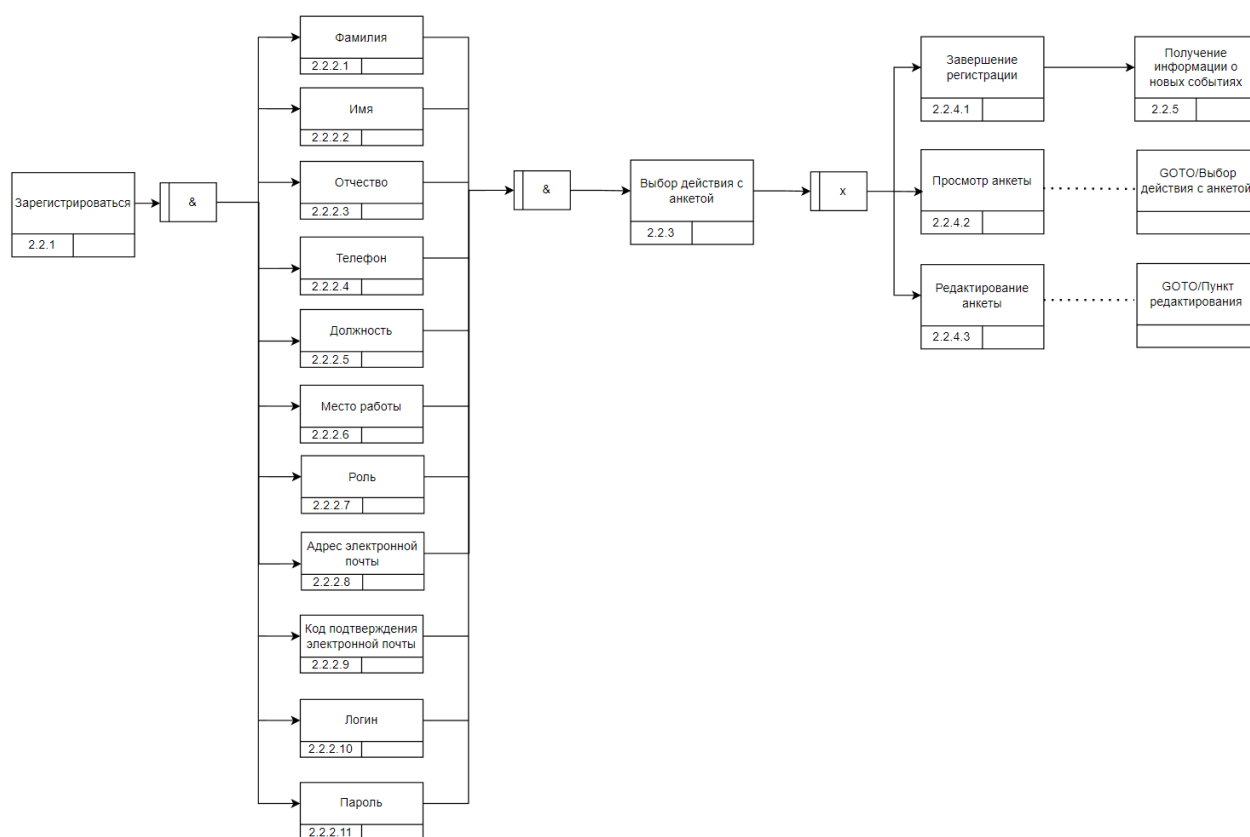


Рисунок 8 – Декомпозиция блока «Зарегистрироваться»

После нажатия кнопки «Зарегистрироваться», чат-бот будет предлагать ввести необходимые данные по очереди от блока 2.2.2.1 до 2.2.2.11. Когда пользователь заполнит анкету, чат-бот предложит несколько вариантов, показанных в блоках 2.2.4.1 – 2.2.4.3. Завершение регистрации отправляет информацию об анкете в веб-приложение и, в случае одобрения администратора, пользователь сможет пользоваться веб-приложением и получать оповещения через чат-бота и электронную почту; редактирование



анкеты позволяет отредактировать определенный пункт в блоках 2.2.2.1 - 2.2.2.11; просмотр анкеты выводит информацию об анкете пользователю.

На рисунке 9 представлена декомпозиция блока авторизации.

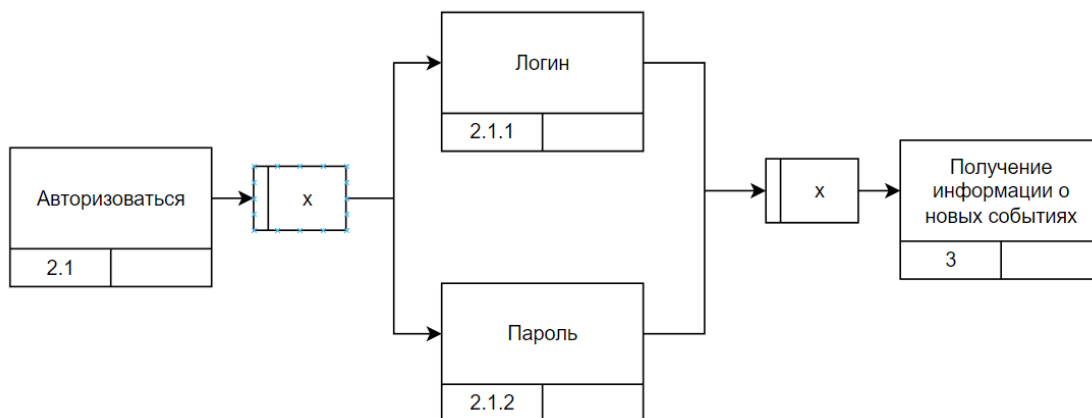


Рисунок 9 – Декомпозиция блока «Авторизоваться»

В данном блоке чат-бот предлагает ввести логин и пароль по очереди, после чего, в случае удачного ввода, пользователь будет получать информацию о событиях через чат-бота Telegram.

#### 1.4.2 Моделирование потоков данных в нотации DFD

Диаграмма потока данных DFD отражает последовательность процессов и логику взаимодействия элементов рассматриваемой ИС.

На рисунках 10-11 представлены диаграмма DFD веб-приложения и ее декомпозиция.

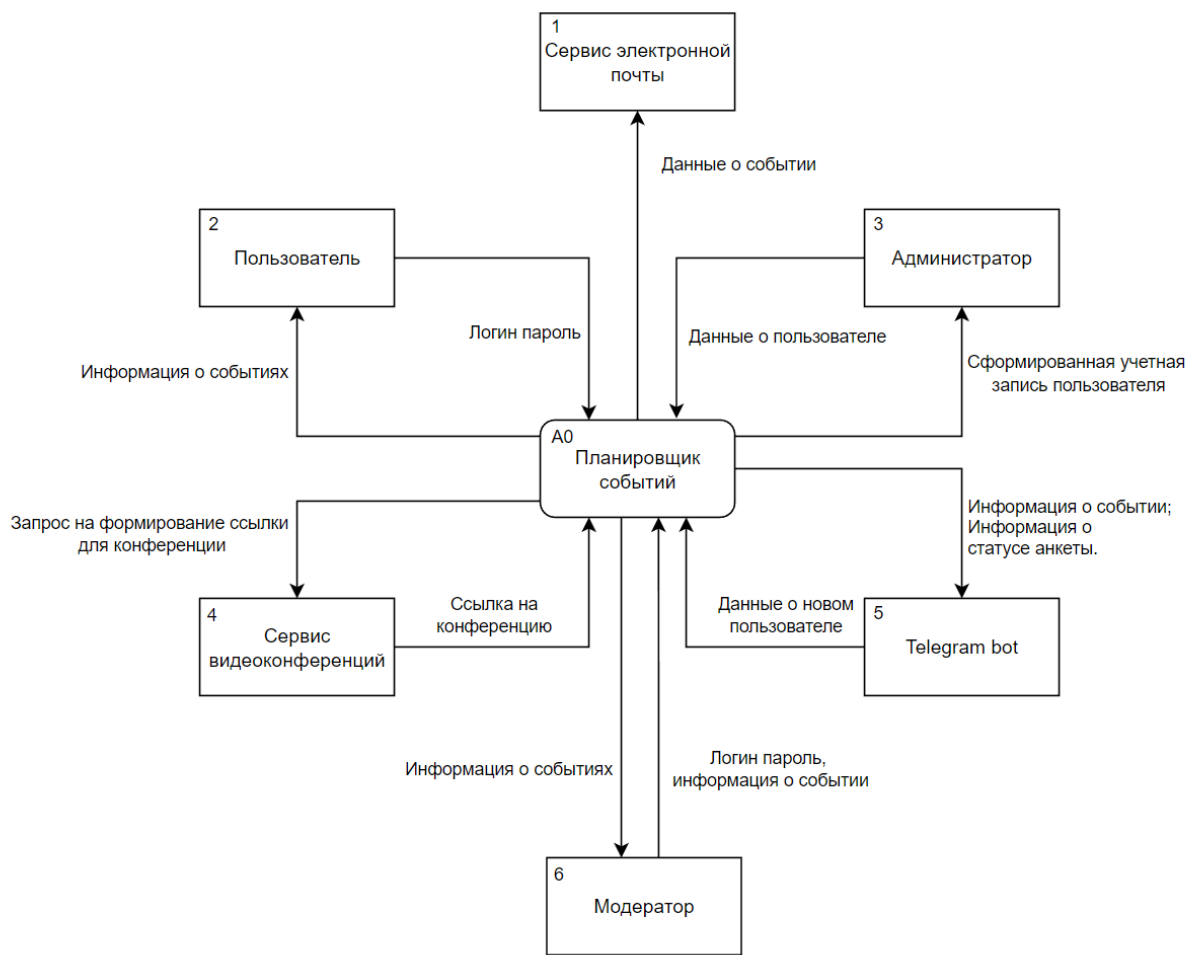


Рисунок 10 – Контекстная диаграмма DFD веб-приложения

Внешними сущностями системы являются: пользователь, администратор, модератор, Telegram bot, сервис видеоконференций, сервис электронной почты.

Система обрабатывает четыре основных процесса: открытие списка событий (блоки A1-A2), добавление нового пользователя (блоки A6-A9), создание события (блоки A3-A5), проверка заявок новых пользователей чат-бота Telegram (блоки A10-A13). Для пользователя каждый этап начинается с авторизации, которая происходит в блоке A1. Здесь система берет информацию из хранилища данных о пользователе и информацию, введенную пользователем, после чего сравнивает ее, затем определяет роль пользователя и, в зависимости от роли, предоставляет доступ к функциям системы.

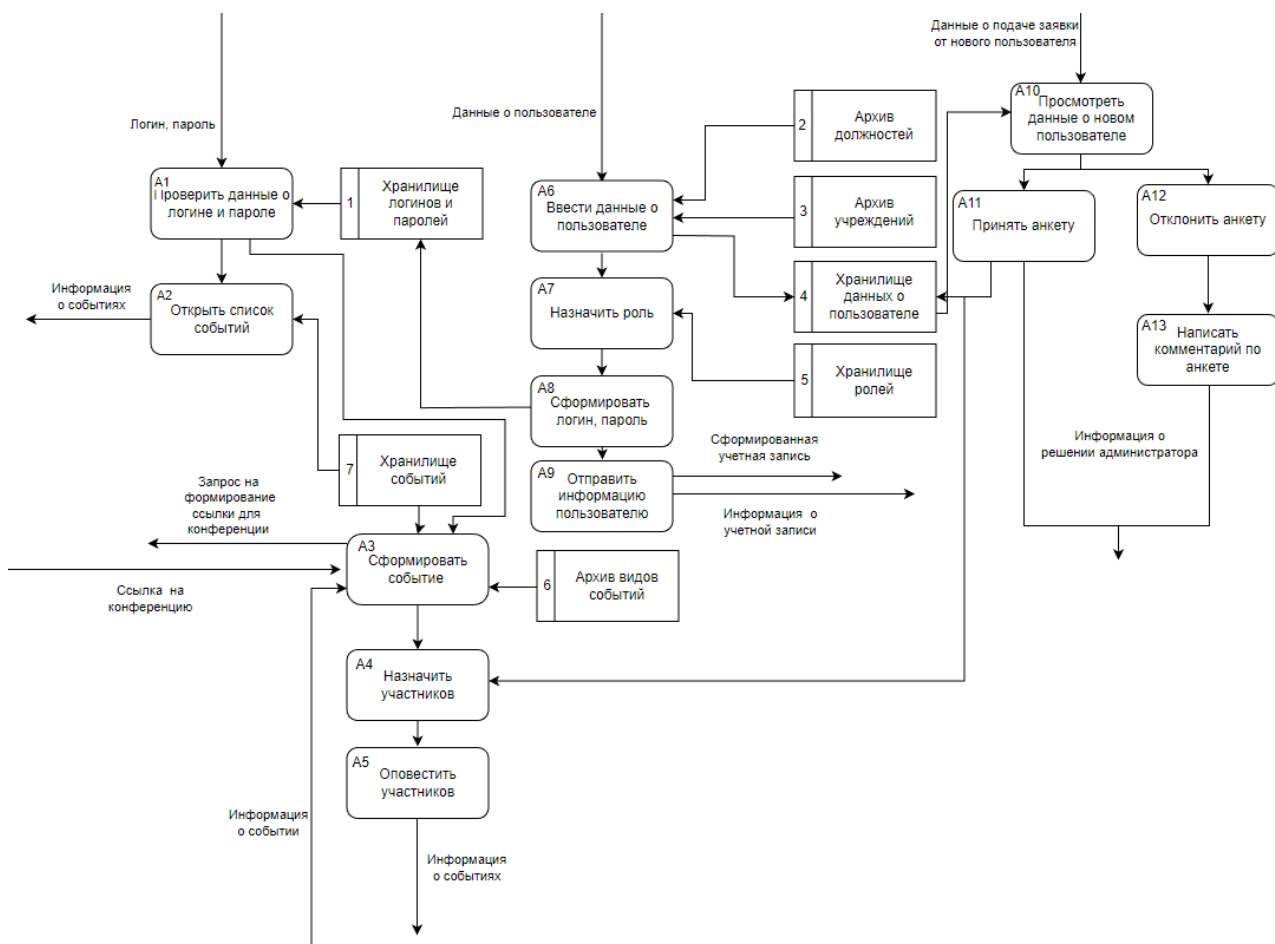


Рисунок 11 – DFD декомпозиция блока «Планировщик событий»

Процесс добавления нового пользователя доступен только администратору. Данный процесс на разных этапах взаимодействует с различными таблицами базы данных. В блоке ввода данных о пользователе (блок А6) система берет информацию, вводимую администратором, а также из архива должностей и архива учреждений и отправляет в хранилище данные о пользователе. В блоке назначения роли (блок А7) берет информацию из хранилища ролей. Этап формирования логина и пароля (блок А8) взаимодействует с хранилищем данных о пользователе, отправляя в данную таблицу информацию о логине и пароле. На стадии отправления информации пользователю (блок А9) ИС не взаимодействует с базой данных. На данном этапе ИС заканчивает формировать учетную запись и оповещает пользователя о создании аккаунта, отправляя информацию об учетной записи посредством Telegram бота.

Процесс создания события также взаимодействует с различными таблицами базы данных. В блоке заполнения формы события (блок А3) система получает информацию от модератора, а также из таблиц базы данных: архив видов событий и хранилище событий. На этапе назначения участников (блок А4), система берет информацию из хранилища событий и хранилища данных о пользователе, после чего формирует список участников события. Этап оповещения участников (блок А5) аналогичен этапу отправления информации пользователю (блок А6), только отправляется информация не об аккаунте, а о событии и отправляется не одному пользователю, а всем участникам события.

Процесс просмотра анкет новых пользователей взаимодействует с хранилищем данных о пользователях, там она берет информацию, введенную пользователем в чат-боте Telegram и, на этапе просмотра анкет (блок А10), предоставляет ее администратору. Администратор решает принять анкету (блок А11) или отклонить (блок А12). В случае принятия анкеты, данные о решении записываются в хранилище пользователей, после чего пользователя оповещают о результате через чат-бота Telegram. В случае отклонения анкеты, администратор вводит причину, и ИС оповещает о результате пользователя через чат-бота Telegram.

На этапе просмотра списка событий (блок А2) система берет информацию из хранилища событий и отправляет список событий пользователю.

На рисунке 12 представлена контекстная диаграмма DFD чат-бота Telegram.

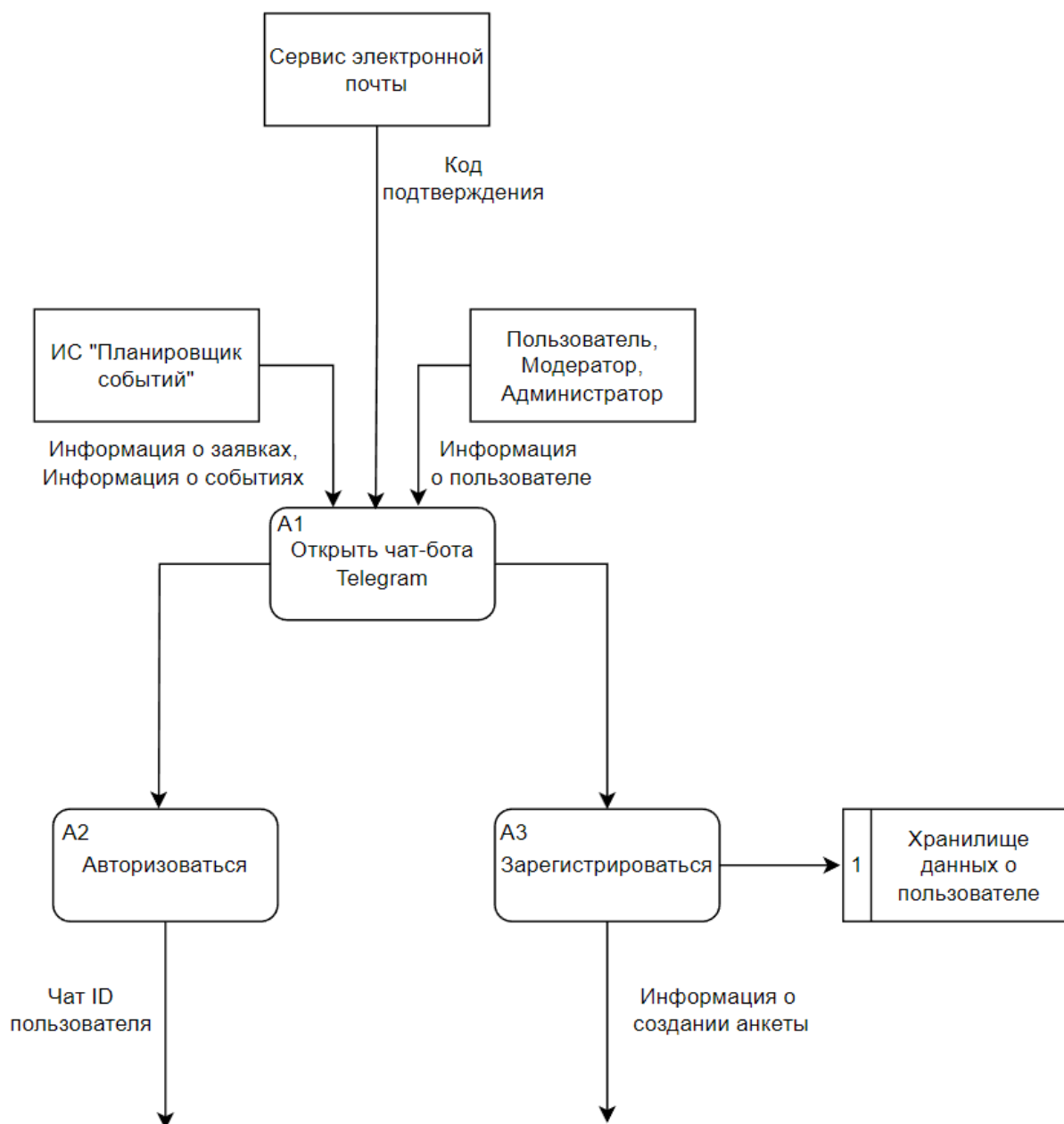


Рисунок 12 – Контекстная диаграмма DFD чат-бота Telegram

Внешними сущностями чат-бота являются: пользователь, администратор, модератор, ИС «Планировщик событий», сервис электронной.

Данный чат-бот позволяет авторизоваться (блок A2) пользователям, которые были добавлены администратором и самостоятельно регистрироваться (блок A3) новым пользователям.

## 1.5 Описание информационно-логической модели данных ER

На рисунке 13 представлена ER модель. Модель состоит из 8 таблиц: пользователи, список участников, события, вид события, тип события, роли, учреждения, должности.

Таблица «Пользователи». Первичным ключом является поле «ID пользователя». Данная таблица хранит в себе всю необходимую информацию о пользователях, включая логин и пароль от аккаунта. Вторичными ключами таблицы являются: ID должности, ID учреждения, ID роли.

Таблица «Список участников». Является связующей таблицей (ассоциативной сущностью), которая связывает таблицы «Пользователи» и «События». Данная таблица позволяет добавлять несколько участников на одно событие.

Таблица «Виды событий». Первичным ключом является поле «ID вида события». Таблица является справочником, который хранит в себе виды событий.

Таблица «События». Первичным ключом является поле «ID события». Данная таблица хранит всю информацию о событиях. Вторичными ключами данной таблицы являются «ID вида события» и «ID типа события».

Таблица «Роли». Первичным ключом является поле «ID роли». Данная таблица является справочником, который хранит в себе роли пользователей, которые определяют уровень доступа к функциям системы.

Таблица «Учреждения». Первичным ключом является поле «ID учреждения». Данная таблица является справочником, который хранит в себе список учреждений, в которых будет функционировать данная ИС.

Таблица «Должности». Первичным ключом является поле «ID должности». Данная таблица является справочником, который хранит в себе список должностей, которые присутствуют в организациях.

Таблица «Тип события». Первичным ключом является поле «ID Типа события». Таблица является справочником, который хранит в себе список

способов проведения событий.

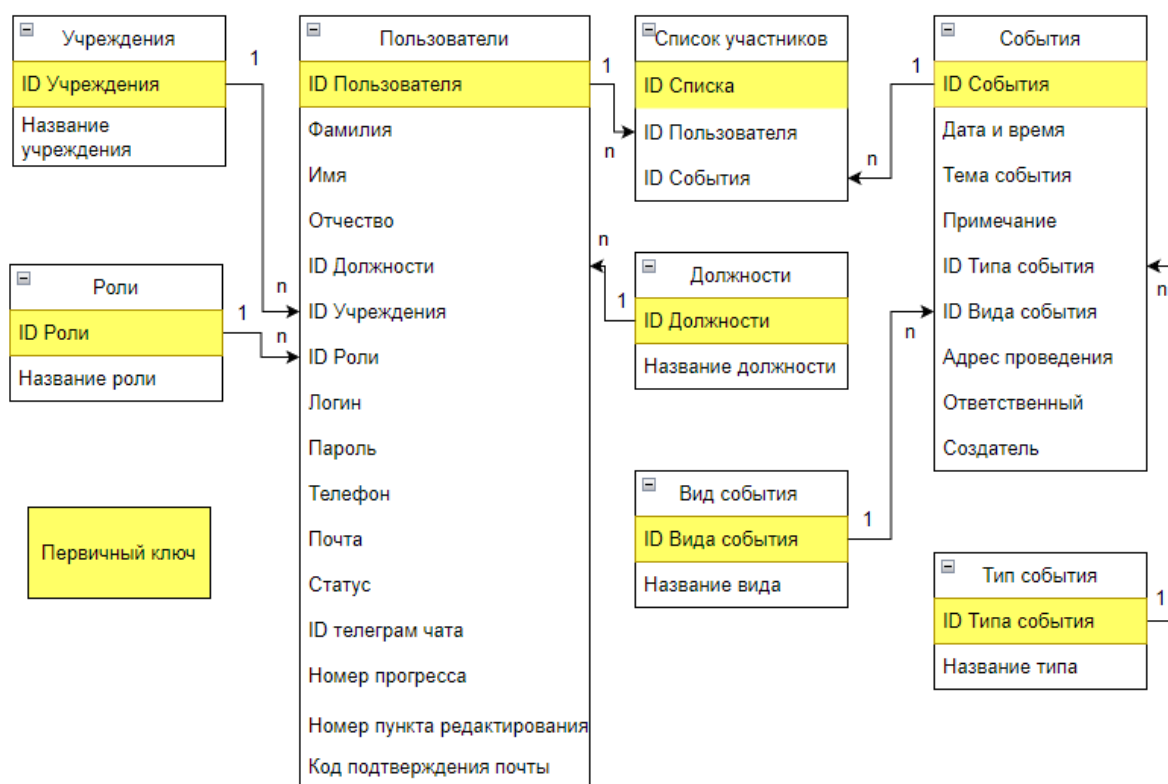


Рисунок 13 – Информационно-логическая модель в духе ER-диаграмм

## 1.6 Анализ средств разработки информационных систем

### 1.6.1 Выбор архитектуры

Разрабатываемая информационная система будет реализована в качестве веб-приложения.

Веб-приложение – клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от

конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными службами. Существенное преимущество построения веб-приложений для поддержки стандартных функций браузера заключается в том, что функции должны выполняться независимо от операционной системы данного клиента. Вместо того, чтобы писать различные версии для Microsoft Windows, Mac OS X, GNU/Linux и других операционных систем, приложение создаётся один раз для произвольно выбранной платформы и на ней разворачивается [5].

Данные преимущества веб-приложения полностью удовлетворяют требованиям Минздрава Хакасии.

### **1.6.2 Выбор языка программирования**

Веб-приложения разрабатываются на специализированных языках программирования. Данные языки программирования должны обеспечивать удобство в написании веб-приложений, высокую производительность, удобство работы с базами данных. Также немаловажным фактором является популярность языка, что обеспечит поддержку приложения на протяжении долгого времени и не затруднит поиск специалистов для этого.

Для сравнения были выбраны следующие языки программирования: Python, Ruby, C#.

Python – высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ.

Недостатками языка являются зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как C или C++, C# [6].



Язык не подходит для разработки данного веб-приложения, так как является довольно ресурсоемким и при этом не обеспечивает высокой производительности.

Ruby – динамический императивный объектно-ориентированный язык программирования. Ruby был создан под влиянием таких языков, как Perl, Eiffel и Smalltalk.

Язык Ruby используется в веб-разработке в составе открытого веб-фреймворка Rails, чаще называемого Ruby on Rails (RoR).

Преимущества:

- открытая разработка;
- работает на многих платформах;
- может внедряться в HTML-разметку;
- относится к языкам программирования сверхвысокого уровня (VHLL), то есть обладает высоким уровнем абстракции и предметным подходом в реализации алгоритмов;
- реализует концептуально чистую объектно-ориентированную парадигму;
- предоставляет продвинутые методы манипуляции строками и текстом.

Недостатки:

- обучение языку выше начального уровня может оказаться непростым;
- информационных ресурсов, посвящённых Ruby, явно недостаточно;
- Ruby менее производителен по сравнению со многими другими языками, применяемыми в веб-разработке;
- Ruby относительно медленно разрабатывается и развивается.

Данный язык не подходит для разработки веб-приложения, так как менее производителен в сравнении с аналогами, а также отстает от выбранных языков в развитии и популярности.

C# – универсальный высокоуровневый многопарадигмальный язык программирования общего назначения. C# охватывает статическую типизацию,

строгую типизацию, лексически ограниченную, императивную, декларативную, функциональную, универсальную, объектно-ориентированную (основанную на классах) и компонентно-ориентированную дисциплины программирования [8].

Преимущества:

– мощная библиотека классов. .NET представляет единую для всех поддерживаемых языков библиотеку классов (какое бы приложение ни приходилось писать на C# – текстовый редактор, чат или сложный веб-сайт – задействуется библиотека классов .NET);

– автоматическая «сборка мусора». Это значит, что в большинстве случаев не придётся заботиться об освобождении памяти;

– высокая производительность, так как входит в класс языков C;

– является пятым по популярности языком программирования, что обеспечивает долгую поддержку языка.

Недостатки:

– приоритетная ориентированность на платформу Windows;

– язык бесплатен только для небольших фирм, индивидуальных программистов, стартапов и учащихся.

Так как данное приложение разрабатывается индивидуальным программистом и является веб-приложением, то выделенные недостатки не повлияют на требования Минздрава Хакасии. Преимущества же языка полностью удовлетворяют определенным ранее требованиям, что позволяет сделать вывод о том, что язык подходит для разработки данного веб-приложения.

### **1.6.3 Выбор системы управления базами данных (СУБД)**

Для полного функционирования данного веб-приложения необходимо создать и подключить базу данных. База данных должна быть создана на профессиональной СУБД, для обеспечения корректной работы приложения.

Для сравнения были выбраны следующие системы управления базами

данных: MySQL, PostgreSQL, MongoDB.

MySQL – одна из самых популярных СУБД для веб-приложений. Фактически, является стандартом для веб-серверов, которые работают под управлением операционной системы Linux. MySQL – это бесплатный пакет программ, однако новые версии выходят постоянно, расширяя функционал и улучшая безопасность. Существуют специальные платные версии, предназначенные для коммерческого использования. В бесплатной версии наибольший упор делается на скорость и надежность, а не на полноту функционала, который может стать и достоинством, и недостатком – в зависимости от области внедрения.

Преимущества:

- распространяется бесплатно;
- прекрасно документирована;
- предлагает много функций, даже в бесплатной версии;
- пакет MySQL включен в стандартные репозитории наиболее распространённых дистрибутивов операционной системы Linux, что позволяет устанавливать её элементарно просто;
- поддерживает набор пользовательских интерфейсов;
- может работать с другими базами данных, включая DB2 и Oracle.

Недостатки:

- требуется потратить много времени и усилий, чтобы заставить MySQL выполнять несложные задачи, хотя другие системы делают это автоматически, например:
  - создавать инкрементные резервные копии;
  - отсутствует встроенная поддержка xml или olap;
  - для бесплатной версии доступна только платная поддержка.

Бесплатная система, которая имеет коммерческую версию – MongoDB. Считается одним из классических примеров NoSQL-систем, использует JSON-подобные документы и схему базы данных. Написана на языке C++. Она предназначена для приложений, которые используют как структурированные,

так и неструктурированные данные. Ядро является очень гибким и работает при подключении базы данных к приложениям через драйверы MongoDB. Существует широкий выбор доступных драйверов, поэтому легко найти драйвер, который будет работать с требуемым языком программирования.

Поскольку изначально система MongoDB не была разработана для обработки моделей реляционных данных (хотя может это выполнять), могут возникнуть проблемы производительности, если попытаться использовать её таким образом. Однако, движок предназначен для обработки различных данных, которые нельзя отнести к реляционным, и может хорошо справляться там, где другие движки работают медленно или вообще бессильны.

Преимущества:

- скорость и простота в использовании;
- данные любой структуры могут быть сохранены/прочитаны быстро и легко.

Недостатки:

- SQL не используется в качестве языка запросов;
- инструменты для перевода SQL-запросов в MongoDB доступны, но их следует рассматривать именно как дополнение;
- программа установки может занять много времени.

PostgreSQL является одним из нескольких бесплатных популярных вариантов СУБД, часто используется для ведения баз данных веб-сайтов. Это весьма старая система, поэтому в настоящее время она хорошо развита, и позволяет пользователям управлять как структурированными, так и неструктурированными данными. Может быть использована на большинстве основных платформ, включая Linux (где особенно хорошо проявляется производительность). Прекрасно справляется с задачами импорта информации из других типов баз данных с помощью собственного инструментария.

Движок БД может быть размещен в ряде сред, в том числе виртуальных, физических и облачных. Самая свежая версия PostgreSQL предлагает обработку больших объемов данных и увеличение числа одновременно работающих

пользователей. Безопасность была улучшена благодаря поддержке DBMS\_SESSION.

Преимущества:

–является масштабируемым решением и позволяет обрабатывать терабайты данных;

–поддерживает формат json;

–существует множество predefined функций;

–доступен ряд интерфейсов.

Недостатки:

– документация туманна, поэтому, возможно, ответы на некоторые вопросы придется искать в интернете;

– скорость работы может падать во время проведения пакетных операций или выполнения запросов чтения [8].

Проведя анализ выбранных СУБД, было решено использовать в проекте СУБД PostgreSQL. Несмотря на свои недостатки, данная СУБД является более подходящей среди рассматриваемых СУБД, так как она самая продвинутая и функциональная, является бесплатной как в использовании, так и в поддержке, также она позволяет использовать язык SQL в качестве языка запросов. Данные критерии являются наиболее значимыми при выборе СУБД.

### **1.7 Выводы по разделу: «Анализ предметной области. Выбор средств проектных решений»**

В процессе анализа предметной области была проанализирована основная деятельность Минздрава Хакасии и его подведомственного учреждения ГБУЗ РХ «РМИАЦ», изучен процесс планирования событий, таких как заседания, совещания, собрания, мероприятия, конференции и т.п.

Перед началом проектирования были проанализированы уже имеющиеся программные продукты на рынке, подходящие для автоматизации процесса планирования событий. Анализ выявил, что для внутреннего процесса

необходимо разработать собственную информационную систему, поскольку ГБУЗ РХ «РМИАЦ» работает с конфиденциальной информацией.

Для проектирования системы было решено построить модель действий пользователя в системе в нотации IDEF3 и модель потоков данных в нотации DFD. Данные модели позволяют по-разному взглянуть на систему, оценить объем работы до разработки.

Перед разработкой было проведено сравнение популярных сред и средств для разработки информационных систем. По результатам сравнения было решено выбрать язык программирования C#, среду разработки Visual Studio.

Была разработана информационно-логическая модель БД. В качестве системы управления базами данных была выбрана PostgreSQL.

## **2 Описание разработки информационной системы планирования событий Министерства здравоохранения Республики Хакасия**

### **2.1 Описание разработки базы данных для информационной системы планирования событий Министерства здравоохранения Республики Хакасия**

Для начала необходимо установить pgAdmin4, создать БД и добавить в нее таблицы. Созданная БД представлена на рисунке 14.

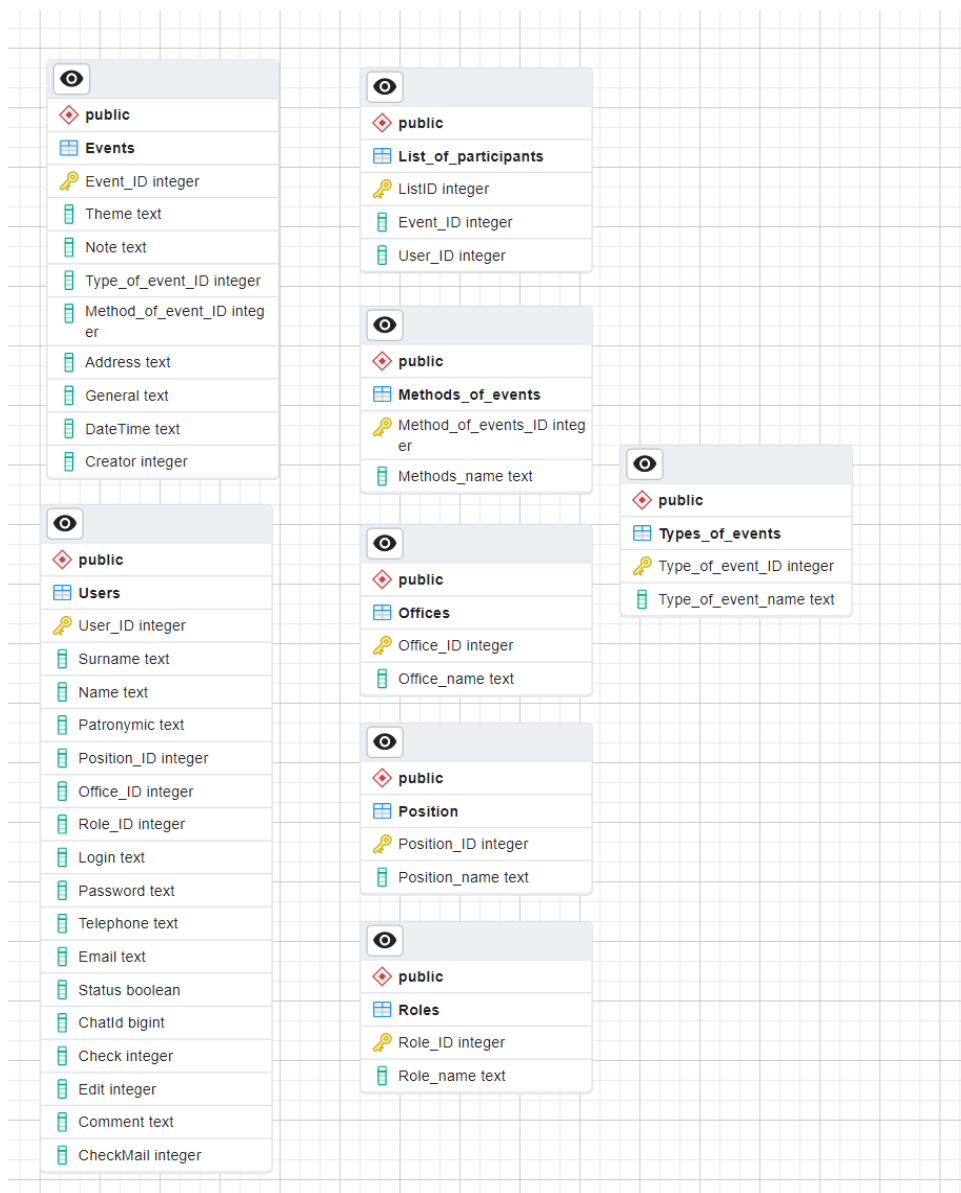


Рисунок 14 – Таблицы созданной БД

Перед подключением к БД, необходимо вручную заполнить таблицы «Offices», «Position», «Roles», «Type of events» и «Methods of events».

Заполненные таблицы показаны на рисунках 15-19.

	Role_ID [PK] integer	Role_name text
1	1	Пользователь
2	2	Администратор
3	3	Модератор

Рисунок 15 – Таблица «Roles»

	Type_of_event_ID [PK] integer	Type_of_event_name text
1	1	Рабочее совещание
2	2	Планерное совещание с Министром
3	3	Коллегия
4	4	Семинар

Рисунок 16 – Таблица «Type of events»

Список должностей взят из реестра справочников [10].

	Position_ID [PK] integer	Position_name text
1	30	Главный врач (начальник, директор)
2	31	Заместитель руководителя (начальника, директора)
3	32	Заведующий (главный врач, начальник) структурного подразделен...
4	33	Главная медицинская сестра
5	34	Врач-специалист
6	35	Средний медицинский персонал
7	36	ИТ-специалист
8	37	Бухгалтер
9	38	Специалист отдела кадров
10	39	Регистратор(оператор колл-центра)
11	40	Секретарь руководителя
12	41	Иные профессии рабочих и служащих

Рисунок 17 – Таблица «Position»

	Method_of_events_ID [PK] integer	Methods_name text
1	1	Видеовстреча
2	2	Очная встреча

Рисунок 18 – Таблица «Methods of events»

Список медицинских организаций Республики Хакасия взят с официального сайта Министерства здравоохранения Республики Хакасия [11].

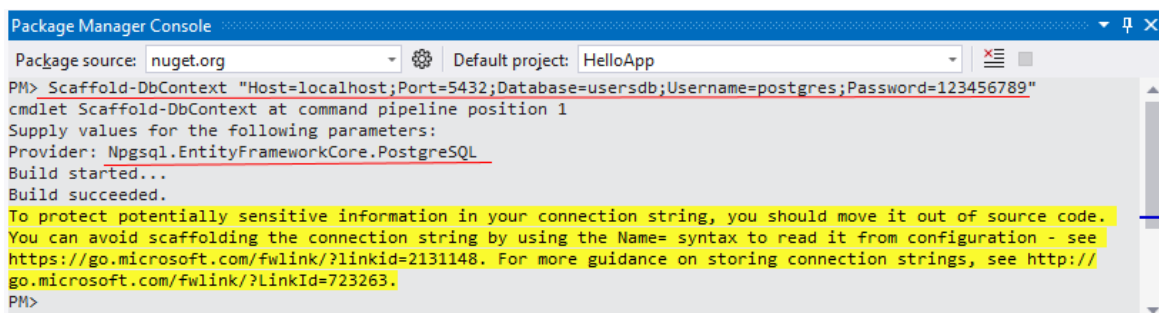


	Office_ID [PK] integer	Office_name text
1	1	ГБУЗ РХ «Абазинская городская больница»
2	2	ГБУЗ РХ «Абаканская городская клиническая станция скорой медицинской помощи»
3	3	ГБУЗ РХ «Абаканская межрайонная клиническая больница»
4	4	ГБУЗ РХ «Аскизская межрайонная больница»
5	5	ГБУЗ РХ «Бейская районная больница»
6	6	ГБУЗ РХ «Белоярская районная больница»
7	7	ГБУЗ РХ «Боградская районная больница»
8	8	ГБУЗ РХ «Копьевская районная больница»
9	9	ГБУЗ РХ «Республиканская детская клиническая больница»
10	10	ГБУЗ РХ «Республиканская клиническая больница им. Г.Я. Ремишевской»
11	11	ГБУЗ РХ «Республиканская клиническая инфекционная больница»
12	12	ГБУЗ РХ «Республиканская клиническая офтальмологическая больница имени Н. М. Одежкина»
13	13	ГБУЗ РХ «Республиканская клиническая психиатрическая больница»
14	14	ГКУ РХ «Учреждение централизованного обслуживания Минздрав РХ и подведомственных ему учреждени...
15	15	ГБУ РХ «Ресфармация»
16	16	ГКУЗ РХ «Республиканское клиническое бюро судебно-медицинской экспертизы»
17	17	ГКУЗ РХ «Республиканский центр общественного здоровья и медицинской профилактики»
18	18	ГКУЗ РХ «Республиканский центр медицины катастроф»
19	19	ГКУЗ РХ «Республиканский центр крови»
20	20	ГКУЗ РХ «Республиканский медицинский центр мобилизационных резервов «Резерв»
21	21	ГБУЗ РХ «Ширинская межрайонная больница»
22	22	ГБУЗ РХ «Черногорский межрайонный родильный дом»
23	23	ГБУЗ РХ «Черногорская межрайонная детская больница»
24	24	ГБУЗ РХ «Черногорская межрайонная больница»
25	25	ГБУЗ РХ «Черногорская городская стоматологическая поликлиника»
26	26	ГБУЗ РХ «Усть-Абаканская районная больница имени Николая Ивановича Солошенко»
27	27	ГБУЗ РХ «Таштыпская районная больница»
28	28	ГБУЗ РХ «Сорская городская больница»
29	29	ГБУЗ РХ «Саяногорская межрайонная больница»
30	30	ГБУЗ РХ «Саяногорская межрайонная больница рабочего поселка Майна»
31	31	ГБУЗ РХ «Саяногорская городская стоматологическая поликлиника»
32	32	ГБУЗ РХ «Саяногорская городская станция скорой медицинской помощи»
33	33	ГБУЗ РХ «Республиканский центр профилактики и борьбы со СПИД»
34	34	ГБУЗ РХ «Республиканский медицинский информационно-аналитический центр»
35	35	ГБУЗ РХ «Республиканский клинический противотуберкулезный диспансер»
36	36	ГБУЗ РХ «Республиканский клинический перинатальный центр»
37	37	ГБУЗ РХ «Республиканский клинический онкологический диспансер»
38	38	ГБУЗ РХ «Республиканский клинический наркологический диспансер»
39	39	ГБУЗ РХ «Республиканский клинический кожно-венерологический диспансер»
40	40	ГБУЗ РХ «Республиканская клиническая стоматологическая поликлиника»

Рисунок 19 – Таблица «Offices»

После того как была создана БД, следует открыть Visual Studio и создать «веб-приложение ASP.NET Core (модель-представление-контроллер)» для веб-приложения и консольное приложение – для чат-бота. Когда приложения созданы, то первым делом к ним необходимо подключить БД. Прежде чем начать подключение БД, необходимо установить nuget-пакет «Npgsql.EntityFrameworkCore.PostgreSQL», чтобы IDE могла нормально взаимодействовать с необходимой БД. Для подключения к существующей базе

данных в PostgreSQL, необходимо в окне «Package Manager Console» выполнить команду «Scaffold-DbContext», которой передается строка подключения и название провайдера, то есть «Npgsql.EntityFrameworkCore.PostgreSQL» (для выполнения этой команды также необходим пакет Microsoft.EntityFrameworkCore.Tools). Сначала вводится команда «Scaffold-DbContext» и строка подключения, затем в консоли появится слово «Provider», после которого надо будет ввести название провайдера «Npgsql.EntityFrameworkCore.PostgreSQL». Пример подключения к БД представлен на рисунке 20.



```
Package Manager Console
Package source: nuget.org Default project: HelloApp
PM> Scaffold-DbContext "Host=localhost;Port=5432;Database=usersdb;Username=postgres;Password=123456789"
cmdlet Scaffold-DbContext at command pipeline position 1
Supply values for the following parameters:
Provider: Npgsql.EntityFrameworkCore.PostgreSQL
Build started...
Build succeeded.
To protect potentially sensitive information in your connection string, you should move it out of source code.
You can avoid scaffolding the connection string by using the Name= syntax to read it from configuration - see
https://go.microsoft.com/fwlink/?linkid=2131148. For more guidance on storing connection strings, see http://
go.microsoft.com/fwlink/?LinkId=723263.
PM>
```

Рисунок 20 – Пример подключения к БД PostgreSQL

После выполнения данной команды IDE сама создаст все необходимые, для взаимодействия с БД, классы, поля и методы. БД необходимо подключить как к чат-боту, так и к веб-приложению. В обоих случаях подключение происходит одинаково.

Когда БД подключена, следует изменить метод получения строки подключения, чтобы обеспечить более надежную защиту БД. Код необходимо изменить так, чтобы строка подключения БД, которая хранит в себе пароль, передавалась из файла конфигурации appsettings.json. На рисунке 21 представлен уже измененный метод.

```

28     protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
29     {
30         if (!optionsBuilder.IsConfigured)
31         {
32             IConfigurationRoot configuration = new ConfigurationBuilder()
33                 .SetBasePath(AppDomain.CurrentDomain.BaseDirectory)
34                 .AddJsonFile("appsettings.json")
35                 .Build();
36             optionsBuilder.UseNpgsql(configuration.GetConnectionString("DefaultConnection"));
37         }
38     }
39
40

```

Рисунок 21 – Метод получения строки подключения к БД

Весь программный код хранится в онлайн сервисе – системе контроля версий GitHub, для обеспечения сохранности кода в случае поломки компьютера. После подключения БД можно переходить к разработке чат-бота и веб-приложения.

## 2.2 Описание разработки чат-бота Telegram для информационной системы планирования событий Министерства здравоохранения Республики Хакасия

Перед тем как пользоваться приложением пользователю необходимо пройти авторизацию, то есть ввести в окно авторизации логин и пароль от своей учетной записи.

Есть 2 способа получить логин и пароль: пройти регистрацию в чат-боте Telegram или получить логин и пароль у администратора, в данном случае пользователю необходимо будет авторизоваться в чат-боте для получения оповещений о событиях через Telegram.

Рассмотрим способ с регистрацией в чат-боте Telegram. Перед разработкой чат-бота Telegram, его необходимо создать с помощью конструктора ботов «BotFather» в самом Telegram и получить «token» бота. На рисунке 22 показан процесс создания чат-бота Telegram, через конструктор ботов «BotFather».

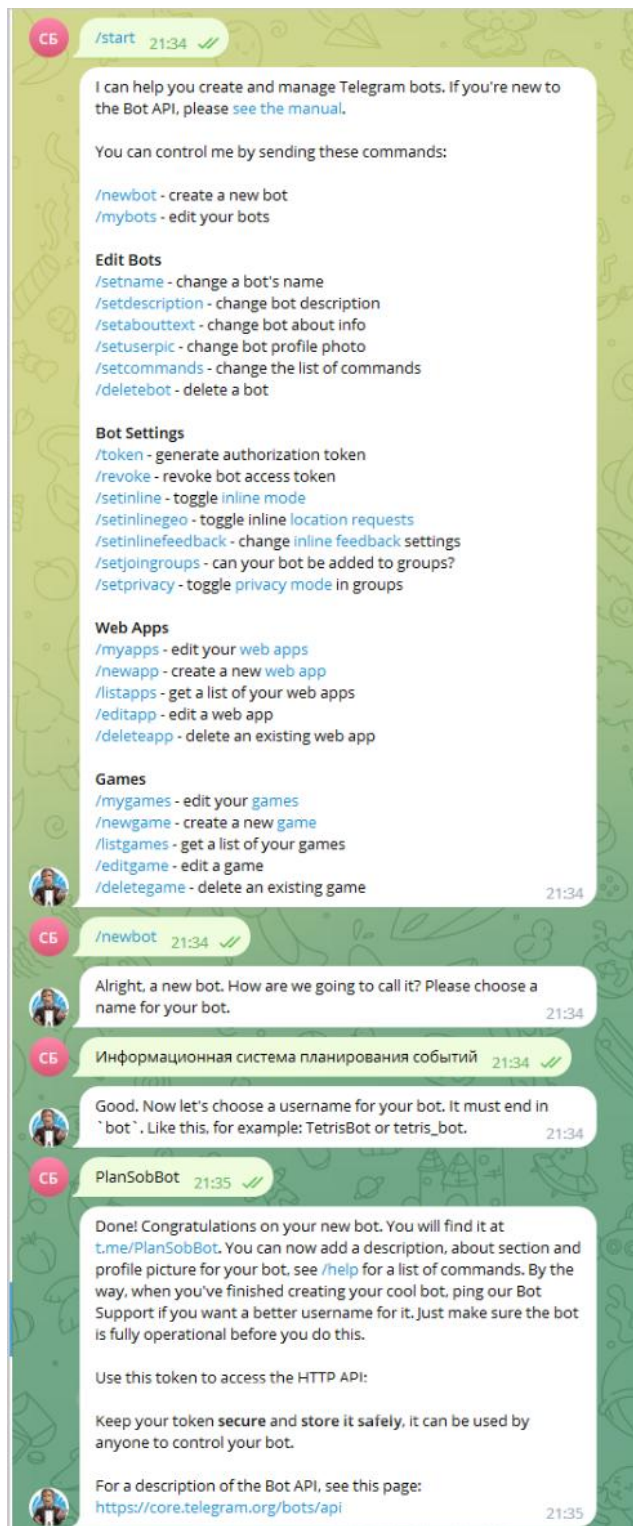


Рисунок 22 – Создание чат-бота Telegram, через конструктор ботов «BotFather»

После создания чат-бота, «BotFather» отправляет «token» бота, который используется в приложении. Данный «token» является секретной информацией, так как он предоставляет полный доступ к боту, соответственно его необходимо хранить в специальном файле, а именно в файле конфигурации. К

данному файлу нельзя получить доступ из сторонних источников, только с того компьютера, на котором он находится, поэтому при выгрузке репозитория GitHub, на другой компьютер, данный файл придется заполнять заново. Данный файл представлен на рисунке 23.

```
1 {
2   "profiles": {
3     "ConsoleApp3": {
4       "commandName": "Project",
5       "environmentVariables": {
6         "BotToken": "6264<74854:AAH",
7         "DefaultConnection": "Host=localhost;Port=;Database=Planirovshik_Sobitiy;Username=postgres;Password:",
8         "SmtpClient": "smtp.ethereal.email",
9         "SmtpPort": "587",
10        "EmailAddress": "brad",
11        "EmailPassword": "EWZ"
12      }
13    }
14  }
15 }
```

Рисунок 23 – Файл конфигурации программы

При открытии чат-бота в первый раз, Telegram предложит ввести стандартную команду для всех ботов «/start». Когда пользователь вводит данную команду, чат-бот должен собрать всю необходимую информацию для взаимодействия с ним, а именно «chatId». «chatId» – это идентификационный номер пользователя, который позволяет отправлять сообщения только тому пользователю, который взаимодействует с чат-ботом. Помимо сбора необходимой информации, бот должен также как-то реагировать на сообщение пользователя, чтобы тот понял, что бот находится в рабочем состоянии. На рисунках 24-26 представлены коды сбора необходимой информации, обработчика команды «/start» и результат, который отправляет обработчик.

```

Ссылка 1
async Task HandleUpdateAsync(ITelegramBotClient botClient, Update update, CancellationToken cancellation)
{
    ConsoleApp3.User nUser = new ConsoleApp3.User();
    /*Создаю переменную БД*/
    PlanirovshikSobitiyContext db = new PlanirovshikSobitiyContext();
    /*Создаю часто используемые переменные для сокращения написания кода*/
    var message = update.Message;

    long? chatId = null;

    string? messageText = null;
    /*Получаю необходимые для работы переменные из сообщения*/
    if (update.Message is not null)
    {
        if (db.Users.FirstOrDefault(u => u.ChatId == update.Message.Chat.Id) is not null)
        {
            nUser = db.Users.FirstOrDefault(u => u.ChatId == update.Message.Chat.Id);
        };
        message = update.Message;
        chatId = message.Chat.Id;
        messageText = message.Text;
        Console.WriteLine($"Received a '{messageText}' message in chat {chatId}.");
    }

    if (update.CallbackQuery is not null)
    {
        chatId = update.CallbackQuery.From.Id;
        if (db.Users.FirstOrDefault(u => u.ChatId == chatId) is not null)
        {
            nUser = db.Users.FirstOrDefault(u => u.ChatId == chatId);
        };
        Console.WriteLine($"Received a '{update.CallbackQuery.Data}' message in chat {chatId}.");
    }
}

```

Рисунок 24 – Код сбора необходимой информации о пользователе

```

/*Обработчик сообщения /start */
if (update.Message is not null && update.Message.Text == "/start" && nUser.Check is null)
{
    if (nUser.ChatId is null)
    {
        nUser.ChatId = chatId;
        nUser.Check = 0;
        db.Users.Add(nUser);
        db.SaveChanges();
    }
    List<List<KeyboardButton>> ListCom = new List<List<KeyboardButton>> { };
    ListCom.Add(new List<KeyboardButton>()
    {
        new KeyboardButton("Зарегистрироваться"),
        new KeyboardButton("Авторизоваться")
    });
    Message sentMessage = await botClient.SendTextMessageAsync(
        chatId: chatId,
        text: "Вас приветствует бот информационной системы планирования событий!" +
            "\nДля продолжения работы выберите из меню необходимую команду.",
        replyMarkup: GetButtons(ListCom),
        cancellationToken: cancellationToken);
    goto end2;
}

```

Рисунок 25 – Код обработки сообщения «/start»

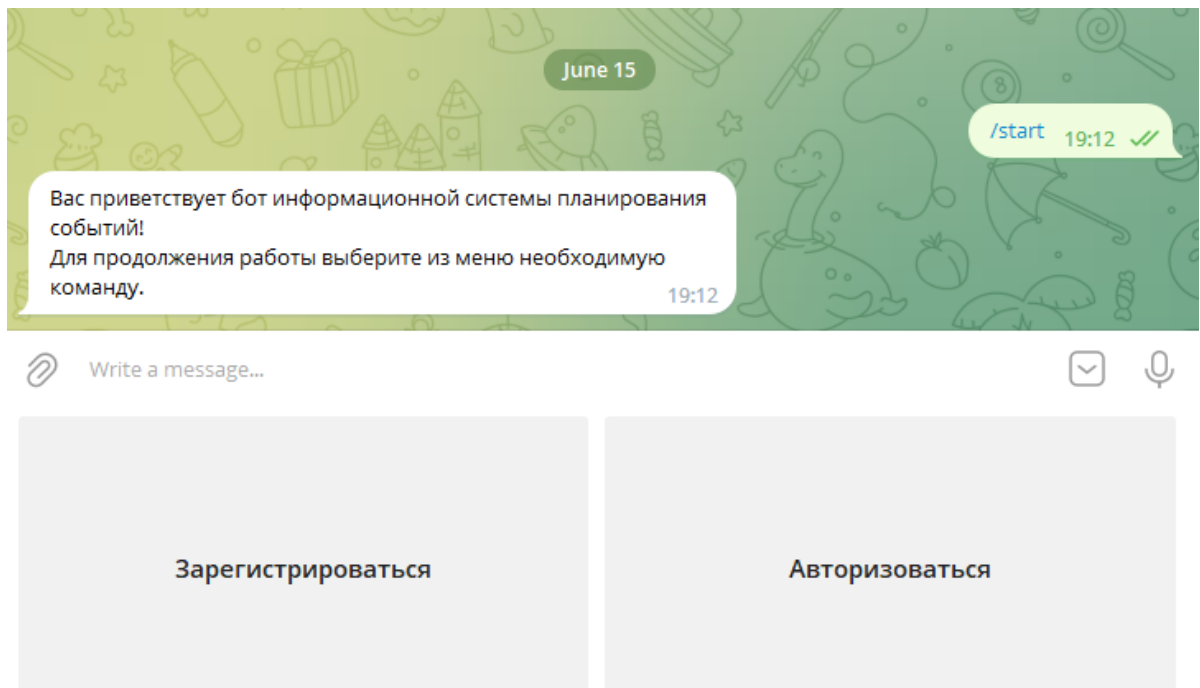


Рисунок 26 – Результат обработки сообщения «/start»

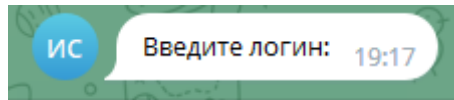
Обработчик, представленный на рисунке 25, работает следующим образом, сначала в БД сохраняется «chatId» пользователя и прогресс заполнения анкеты, затем формируется лист кнопок и, методом `SendMessageAsync`, отправляется сообщение, которое принимает помимо текста, метод формирования кнопок. Большинство последующих обработчиков работает аналогично.

После команды «/start» чат-бот предлагает два варианта действий: зарегистрироваться или авторизоваться.

Авторизация происходит в два пункта, сначала пользователь вводит логин, затем пароль. На каждом этапе ввода данные сверяются с данными в БД. Так как пароль в БД хранится в хэшированном виде, то для сравнения пароля, введенного и хранящегося в БД, необходимо использовать метод библиотеки `BCrypt – Verify`. Способ применения метода показан на рисунке 27. На рисунках 28-29 показаны этапы авторизации.

```
const httpStatusCode = 200;
if (BCrypt.Net.BCrypt.Verify(pass, usr.Password))
{
```

Рисунок 27 – Применение метода `Verify`




 SergeyBov

Рисунок 28 – Этап ввода логина

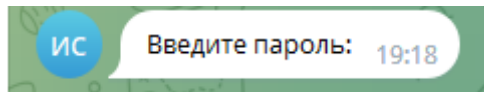


Рисунок 29 – Этап ввода пароля

Так как логин и пароль являются конфиденциальными данными, чат-бот удаляет их специальным методом `DeleteMessageAsync`, данный метод представлен на рисунке 30.

```
await botClient.DeleteMessageAsync(chatId: chatId,  
messageId: update.CallbackQuery.Message.MessageId,
```

Рисунок 30 – Метод удаления сообщения

Регистрация в чат-боте происходит в 2 этапа: заполнение анкеты и проверка анкеты администратором.

На рисунках 31–39 представлены этапы регистрации.

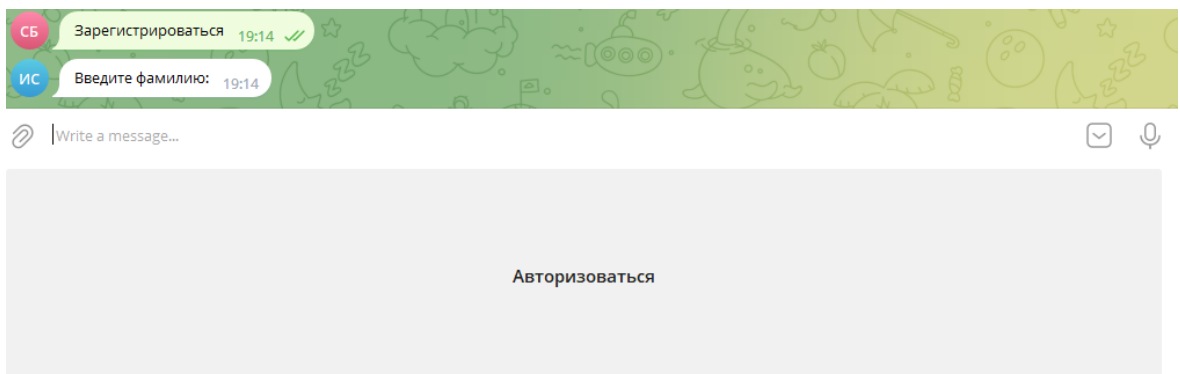


Рисунок 31 – Этап ввода фамилии



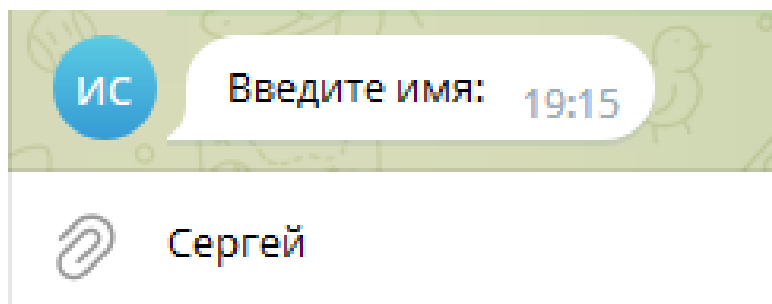


Рисунок 32 – Этап ввода имени

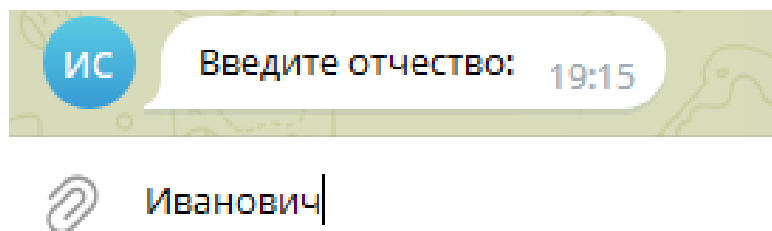


Рисунок 33 – Этап ввода отчества

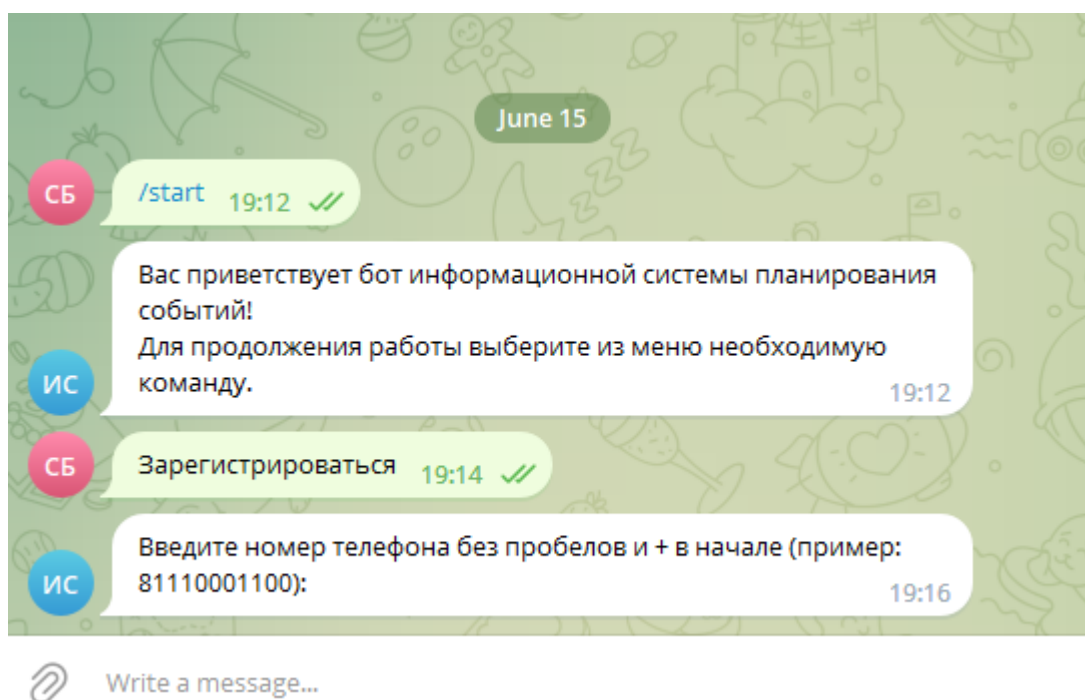
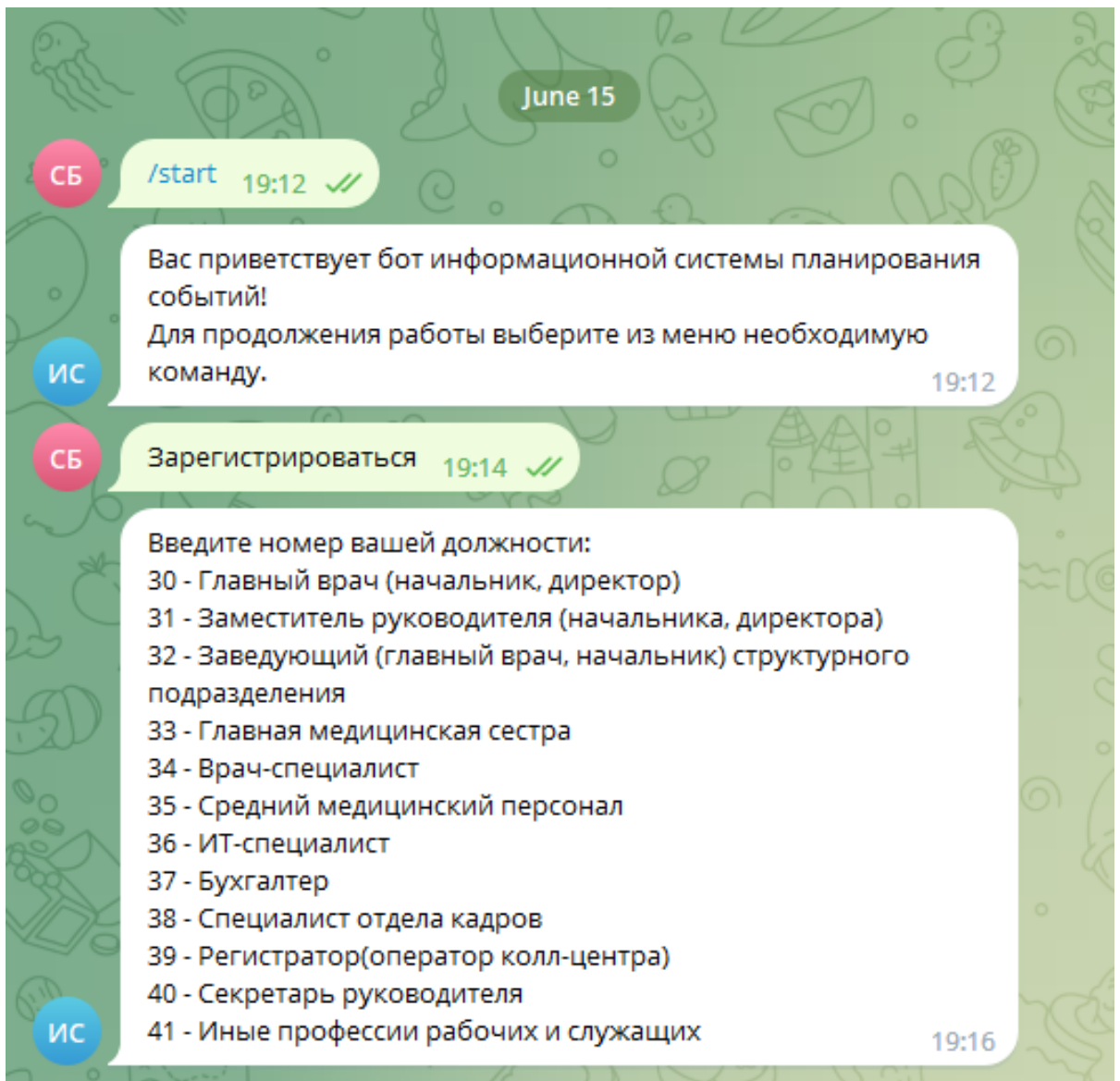


Рисунок 34 – Этап ввода номера телефона



36

Рисунок 35 – Этап ввода должности

Введите номер места работы:

- 1 - ГБУЗ РХ «Абазинская городская больница»
- 2 - ГБУЗ РХ «Абаканская городская клиническая станция скорой медицинской помощи»
- 3 - ГБУЗ РХ «Абаканская межрайонная клиническая больница»
- 4 - ГБУЗ РХ «Аскизская межрайонная больница»
- 5 - ГБУЗ РХ «Бейская районная больница»
- 6 - ГБУЗ РХ «Белоярская районная больница»
- 7 - ГБУЗ РХ «Боградская районная больница»
- 8 - ГБУЗ РХ «Копьевская районная больница»
- 9 - ГБУЗ РХ «Республиканская детская клиническая больница»
- 10 - ГБУЗ РХ «Республиканская клиническая больница им. Г.Я. Ремишевской»
- 11 - ГБУЗ РХ «Республиканская клиническая инфекционная больница»
- 12 - ГБУЗ РХ «Республиканская клиническая офтальмологическая больница имени Н. М. Одежкина»
- 13 - ГБУЗ РХ «Республиканская клиническая психиатрическая больница»
- 14 - ГКУ РХ «Учреждение централизованного обслуживания Минздрав РХ и подведомственных ему учреждений»
- 15 - ГБУ РХ «Ресфармация»
- 16 - ГКУЗ РХ «Республиканское клиническое бюро судебно-медицинской экспертизы»
- 17 - ГКУЗ РХ «Республиканский центр общественного здоровья и медицинской профилактики»
- 18 - ГКУЗ РХ «Республиканский центр медицины катастроф»
- 19 - ГКУЗ РХ «Республиканский центр крови»
- 20 - ГКУЗ РХ «Республиканский медицинский центр мобилизационных резервов «Резерв»
- 21 - ГБУЗ РХ «Ширинская межрайонная больница»
- 22 - ГБУЗ РХ «Черногорский межрайонный родильный дом»
- 23 - ГБУЗ РХ «Черногорская межрайонная детская больница»
- 24 - ГБУЗ РХ «Черногорская межрайонная больница»
- 25 - ГБУЗ РХ «Черногорская городская стоматологическая поликлиника»
- 26 - ГБУЗ РХ «Усть-Абаканская районная больница имени Николая Ивановича Солошенко»
- 27 - ГБУЗ РХ «Таштыпская районная больница»
- 28 - ГБУЗ РХ «Сорская городская больница»
- 29 - ГБУЗ РХ «Саяногорская межрайонная больница»
- 30 - ГБУЗ РХ «Саяногорская межрайонная больница рабочего поселка Майна»
- 31 - ГБУЗ РХ «Саяногорская городская стоматологическая поликлиника»
- 32 - ГБУЗ РХ «Саяногорская городская станция скорой медицинской помощи»
- 33 - ГБУЗ РХ «Республиканский центр профилактики и борьбы со СПИД»
- 34 - ГБУЗ РХ «Республиканский медицинский информационно-аналитический центр»
- 35 - ГБУЗ РХ «Республиканский клинический противотуберкулезный диспансер»
- 36 - ГБУЗ РХ «Республиканский клинический перинатальный центр»
- 37 - ГБУЗ РХ «Республиканский клинический онкологический диспансер»
- 38 - ГБУЗ РХ «Республиканский клинический наркологический диспансер»
- 39 - ГБУЗ РХ «Республиканский клинический кожно-венерологический диспансер»
- 40 - ГБУЗ РХ «Республиканская клиническая стоматологическая поликлиника»

ИС

19:17



Write a message...

Рисунок 36 – Этап ввода организации

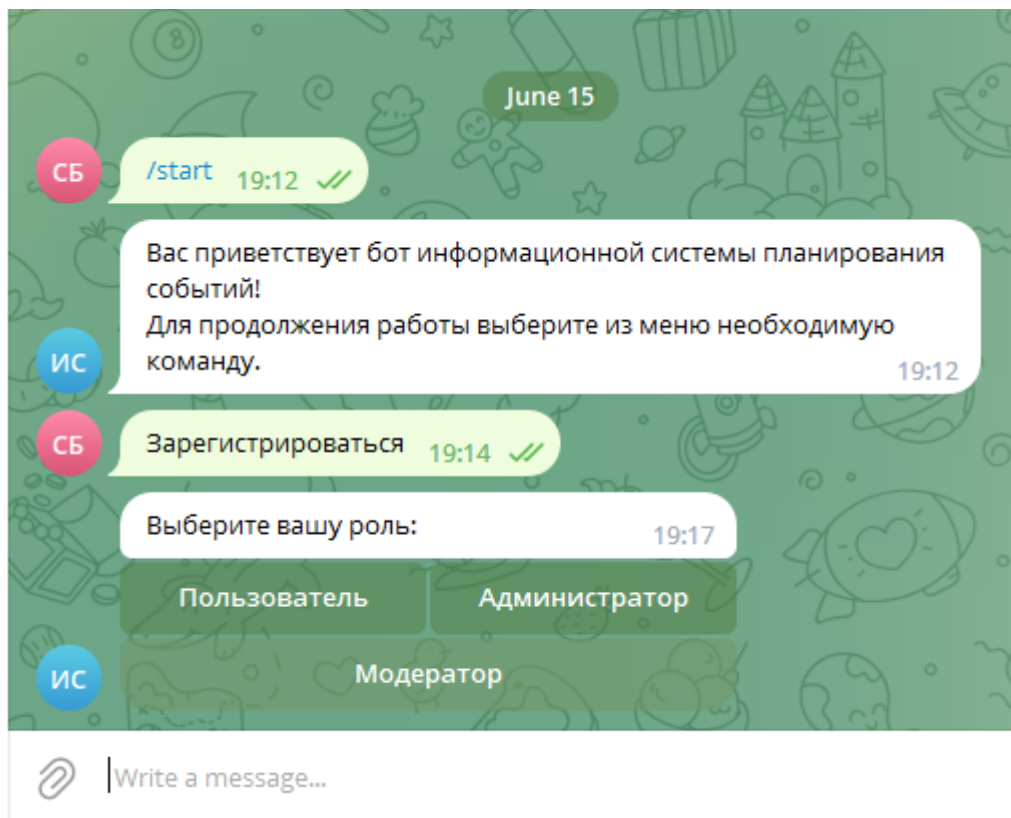


Рисунок 37 – Этап выбора роли

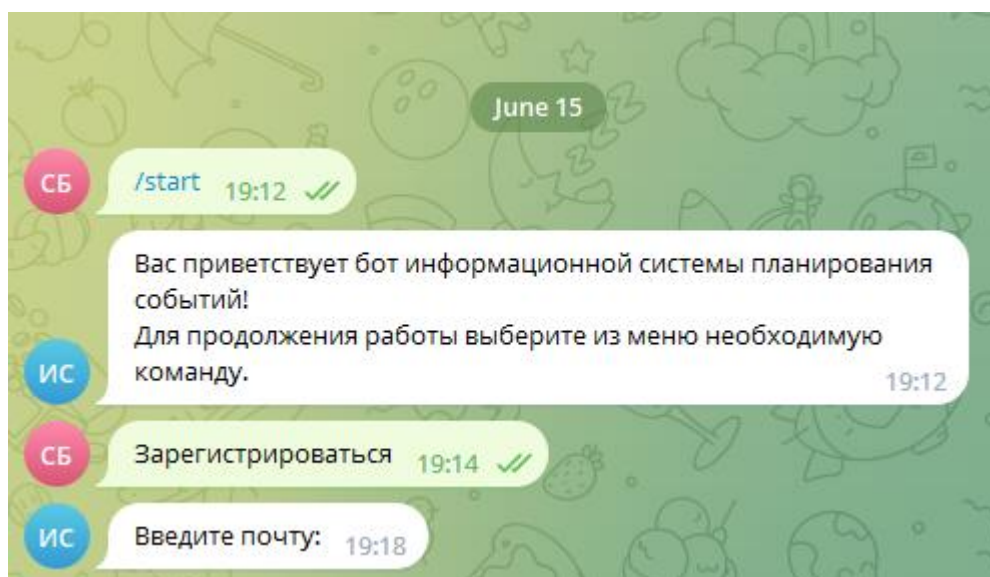


Рисунок 38 – Этап ввода электронной почты

Большинство этапов на программном уровне формируются аналогично обработчику команды «/start», поэтому остановимся на тех этапах, которые

отличаются, а именно: ввод номера телефона, выбор роли, ввод электронной почты.

Отличие на этапе ввода номера телефона заключается в проверке введенных данных. На рисунке 39 представлен код проверки номера телефона.

```
/*Обработчик при вводе номера телефона*/
if ((nUser.Telephone is null && nUser.Check == 4) || nUser.Edit == 4)
{
    /*Обработка при первом вводе*/
    if (message is not null && message.Text is not null && nUser.Telephone is null)
    {
        string telf = message.Text;
        if (telf.Length != 11)
        {
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId - 1,
            cancellationToken: cancellationToken);
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId,
            cancellationToken: cancellationToken);
            Message senMessage = await botClient.SendTextMessageAsync(
            chatId: chatId,
            text: "Внимание! \nНомер телефона должен состоять из 11 символов.",
            replyMarkup: DelKeyboard(),
            cancellationToken: cancellationToken);
            nUser.Telephone = null;
            goto end;
        }
    }
    else
    {
        try
        {
            double tel = Convert.ToDouble(telf);
        }
        catch
        {
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId - 1,
            cancellationToken: cancellationToken);
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId,
            cancellationToken: cancellationToken);
            Message senMessage = await botClient.SendTextMessageAsync(
            chatId: chatId,
            text: "Внимание! \nНомер телефона должен состоять цифр.",
            replyMarkup: DelKeyboard(),
            cancellationToken: cancellationToken);
            goto end;
        }
    }
    ConsoleApp3.User[] telephones = db.Users.ToArray();
    foreach (ConsoleApp3.User a in telephones)
    {
        if (a.Telephone == telf)
        {
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId - 1,
            cancellationToken: cancellationToken);
            await botClient.DeleteMessageAsync(chatId: chatId,
            messageId: message.MessageId,
            cancellationToken: cancellationToken);
            Message senMessage = await botClient.SendTextMessageAsync(
            chatId: chatId,
            text: "Данный телефон уже используется другим пользователем, используйте другой телефон.",
            replyMarkup: DelKeyboard(),
            cancellationToken: cancellationToken);
            goto end;
        }
    }
}
```

Рисунок 39 – Код проверки номера телефона

Данный код проверяет, чтобы номер телефона состоял из 11 символов, состоял только из цифр и не был использован другим пользователем ИС.

Выбор роли отличается тем, что выбор осуществляется не нажатием кнопки на клавиатуре, которая отправляет сообщение, а нажатием кнопки на клавиатуре, которая отправляет запрос. В коде такая клавиатура вызывается методом `InLineKeyboardMarkup`, данные для формирования клавиатуры заполняются в `List<List<InLineKeyboardButton>>`.

Отличие ввода электронной почты заключается в том, что почта, помимо простого ввода, должна быть подтверждена. Подтверждение почты осуществляется отправкой кода подтверждения, который генерируется чат-ботом, на электронный почтовый ящик, введенный пользователем. На рисунке 40 показана отправка кода на почтовый ящик пользователя.

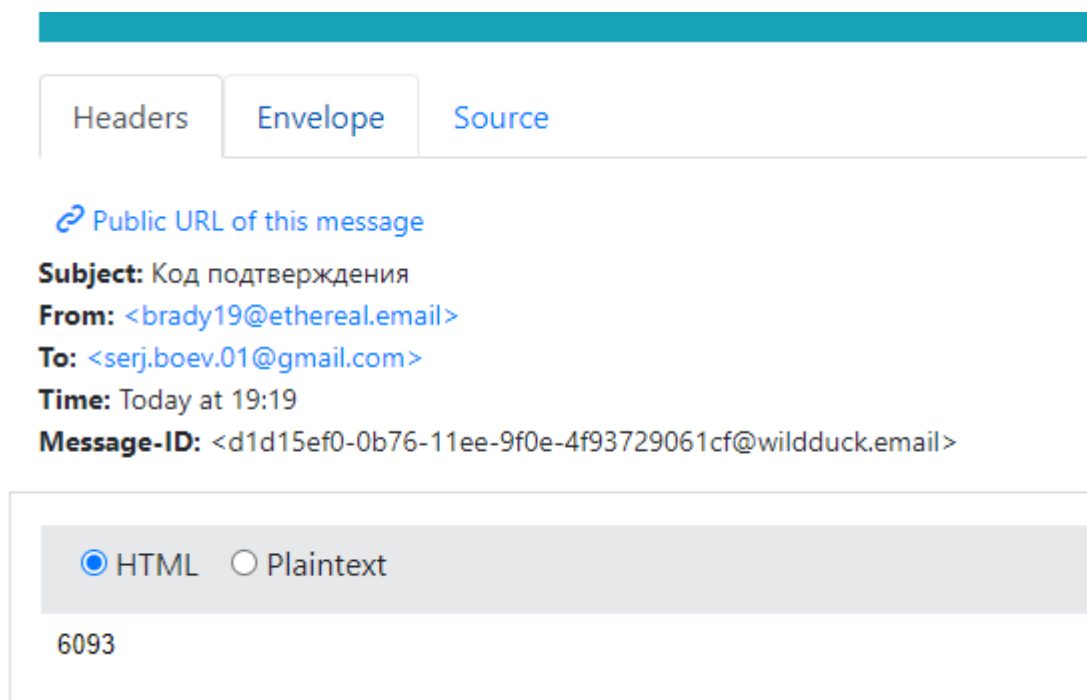


Рисунок 40 – Код подтверждения почты

После заполнения анкеты, бот предлагает завершить формирование анкеты (рисунок 41), но, если вдруг пользователь забыл или допустил ошибку при заполнении, у него также есть возможность отредактировать анкету (рисунок 42) или же посмотреть введенные им данные (рисунок 43).

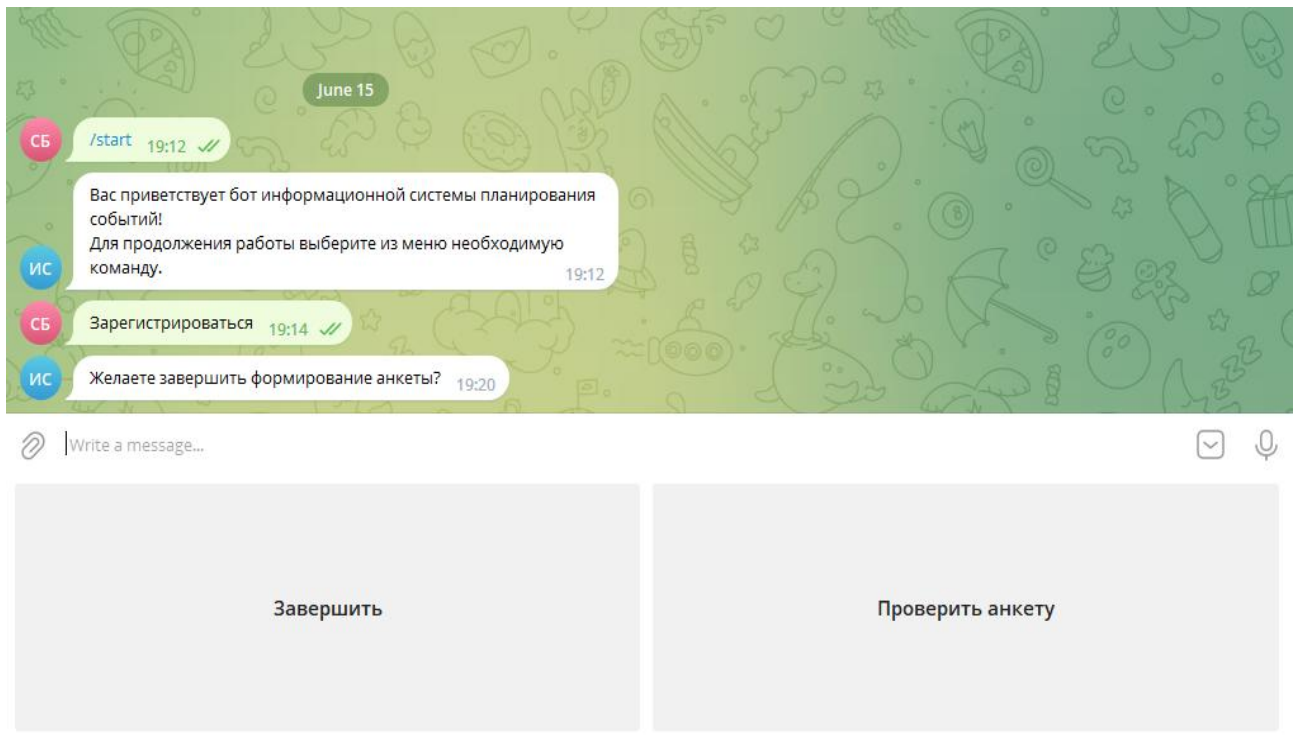


Рисунок 41 – Завершение формирования анкеты

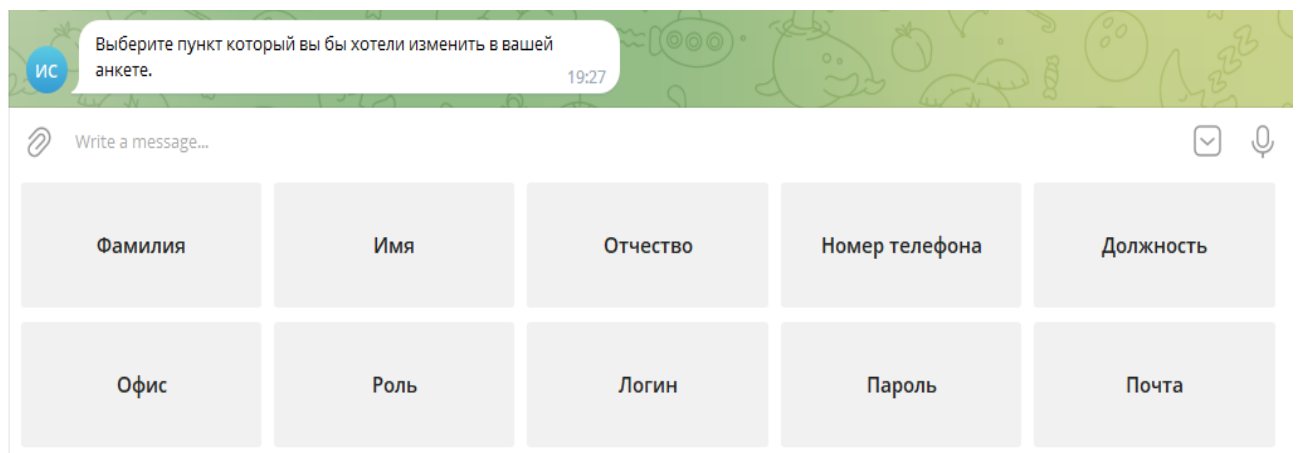


Рисунок 42 – Редактирование анкеты

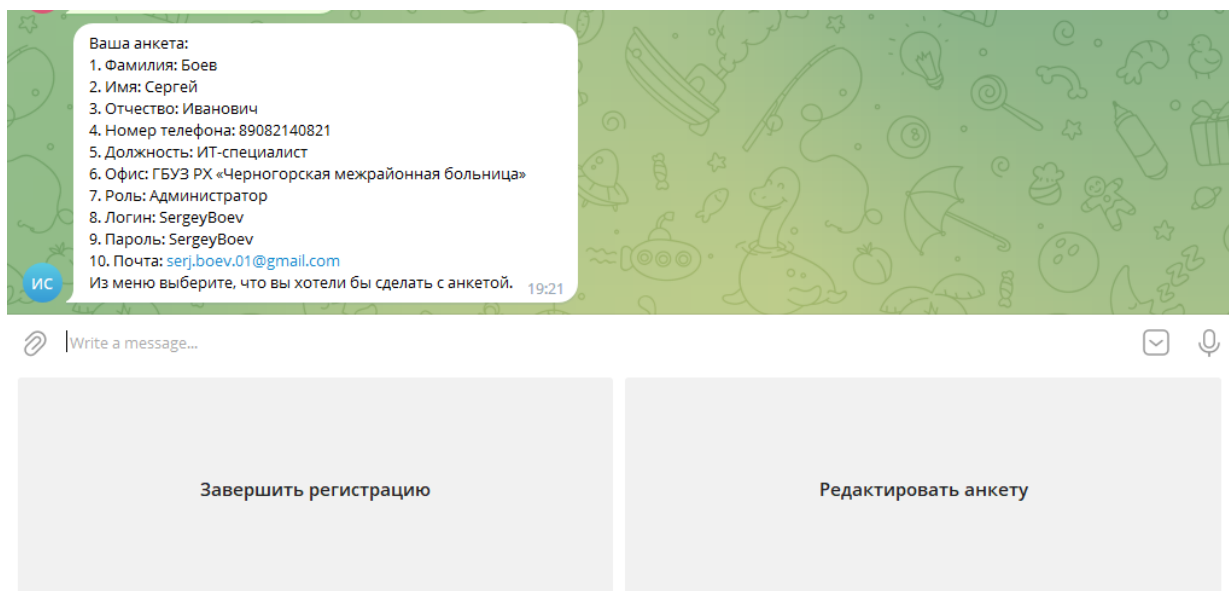


Рисунок 43 – Просмотр анкеты

После отправки анкеты на проверку администратору, бот оповещает об этом пользователя (рисунок 44). Если администратор отклонил анкету, пользователю приходит сообщение от бота об отклонении анкеты с комментарием администратора (рисунок 45), если же анкета была принята, то бот также оповестит об этом пользователя, и тогда пользователь будет получать уведомления от бота о новых событиях (рисунок 46).

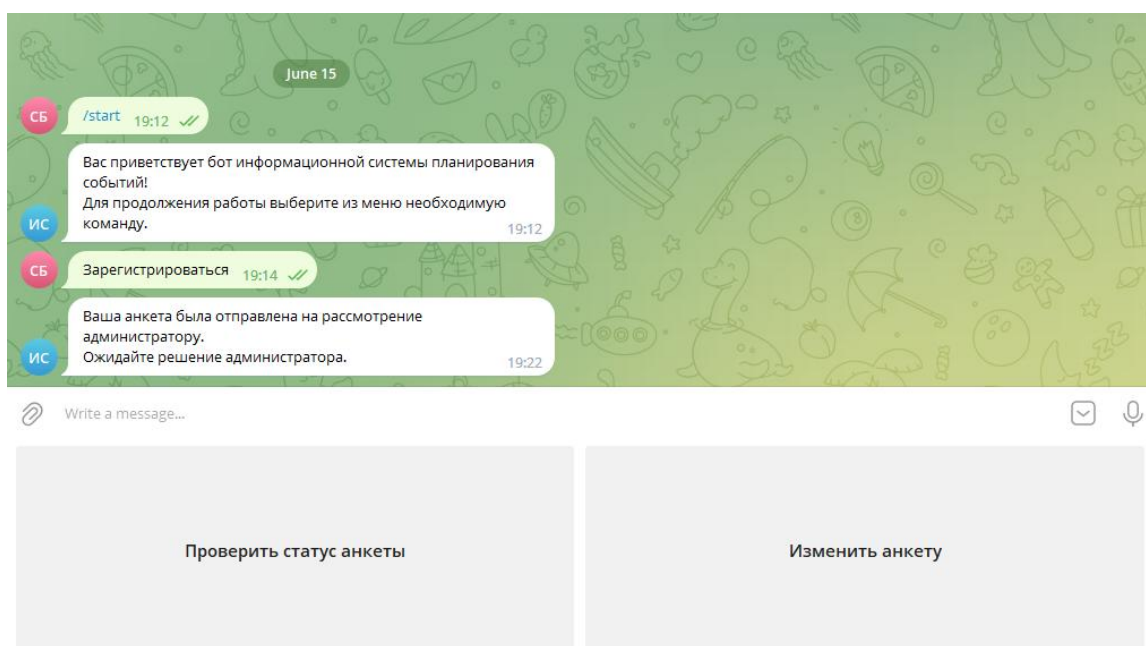


Рисунок 44 – Отправка анкеты на рассмотрение администратору



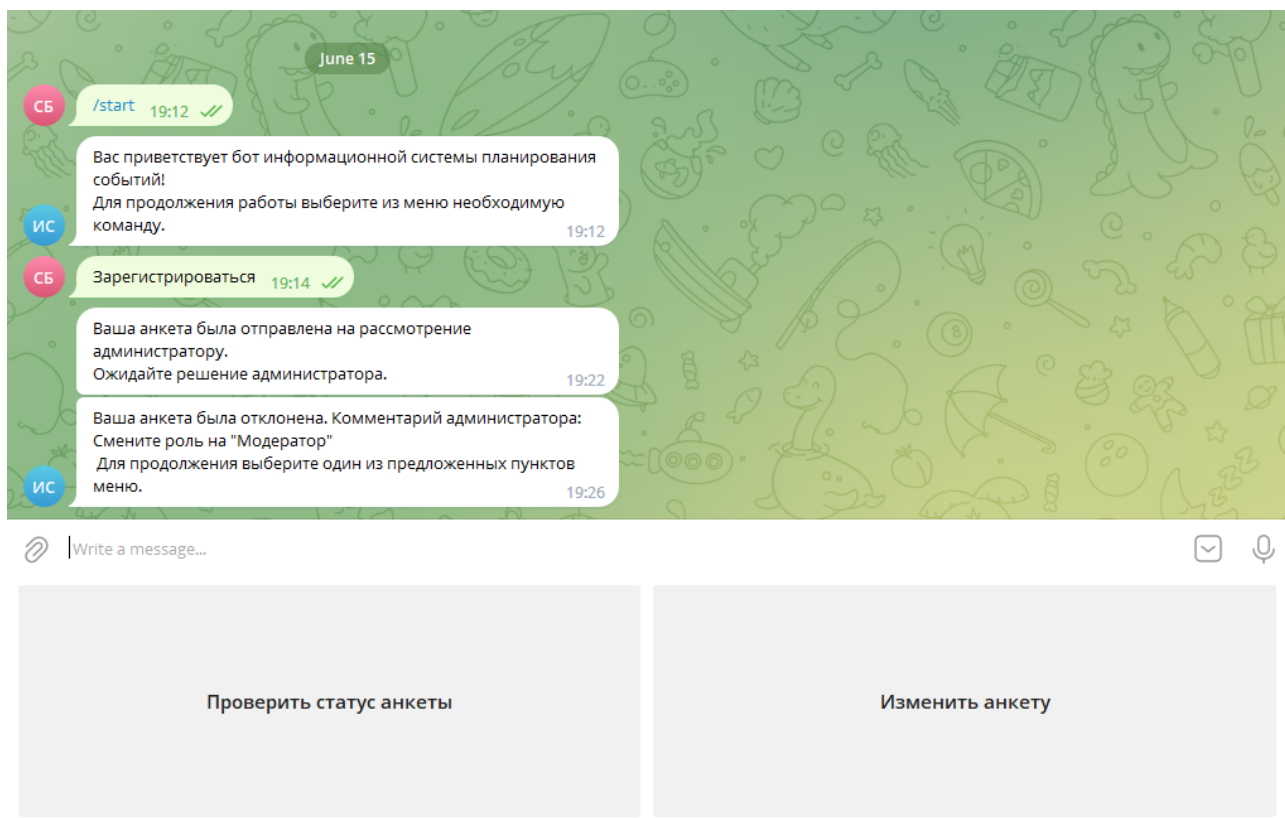


Рисунок 45 – Случай отклонения анкеты

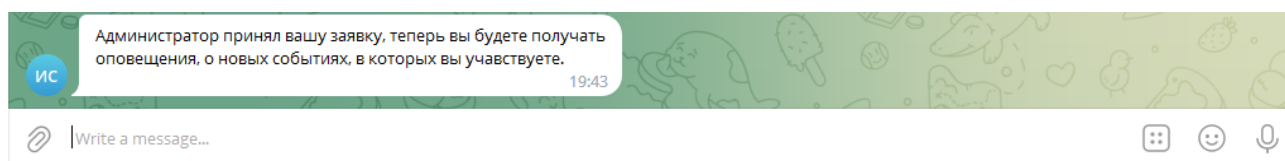


Рисунок 46 – Администратор принял анкету

### 2.3 Описание разработки веб-приложения для информационной системы планирования событий Министерства здравоохранения Республики Хакасия

Рассмотрим второй вариант – получение логина и пароля от администратора. Для того, чтобы администратор мог выдать логин и пароль от учетной записи пользователя, он должен его зарегистрировать. Пароль от учетной записи, как было описано в разработке чат-бота, хэшируется с помощью библиотеки BCrypt. Все поля во всех формах веб-приложения

проверяются на валидность. Форма регистрации пользователя представлена на рисунке 47.

Запланировать мероприятие

Заявки

### Добавить пользователя

Фамилия:

Имя:

Отчество:

Телефон:

Должность:

Главный врач (начальник, дире...

Организация:

ГБУЗ РХ «Абазинская городска...

Роль:

Пользователь

Электронная почта:

Логин:

SergeyVoev

Пароль:

.....

Зарегистрировать

© 2023 - ИС "Планировщик событий"

Рисунок 47 – Форма регистрации пользователя

Теперь, когда у пользователя есть логин и пароль от учетной записи, он может авторизоваться и пользоваться разрешенными ему функциями. Форма авторизации представлена на рисунке 48, также на рисунке показана работа программы проверки валидности введенных данных.

The image shows a light blue rounded rectangular form titled "Авторизация" (Authorization). It contains two input fields: "Логин:" (Login) and "Пароль:" (Password). Below the login field is a red error message "Введите логин" (Enter login). Below the password field is a red error message "Введите пароль" (Enter password). A dark blue button labeled "Войти" (Login) is positioned below the password field. At the bottom of the form, there is a link "Зарегистрироваться через Telegram" (Register via Telegram).

Рисунок 48 – Форма авторизации

Как только пользователь авторизовался, он попадает на главную страницу ИС. На данной странице в верхней части представлено меню навигации по ИС. В центральной части находится календарь мероприятий для навигации по событиям в зависимости от даты. Данный календарь отображает день, который выбрал пользователь и дни, на которые запланированы события. Под календарем отображаются формы, в которых находится информация о событиях.

Информационная система имеет три уровня доступа: пользователь, модератор, администратор.

Пользователь может только просматривать события, на которые он был добавлен.

У модератора, помимо прав пользователя, также есть и права создания, редактирования и удаления событий.

Администратор имеет полный доступ к системе. Помимо функций, доступных модератору, у него есть доступ к регистрации пользователей, рассмотрению заявок с чат-бота Telegram и у него отображаются все события, хранящиеся в системе.

На рисунке 49 изображена главная страница.



Рисунок 49 – Главная страница

Рассмотрим верхнее меню навигации. Календарь мероприятий и добавление пользователя были описаны ранее, поэтому перейдем к пункту «Запланировать мероприятие».

Форма создания события представлена на рисунке 50.

Календарь мероприятий    Добавить пользователя    Запланировать мероприятие    Заявки на регистрацию    Telegram    Выйти

### Запланировать мероприятие

Дата и время: 22.06.2023 09:00

Тема: Планы на неделю

Тип мероприятия: Рабочее совещание

Вид мероприятия: Очная встреча

Примечание: Распределение обязанностей

Место проведения: ЧМБ кабинет 102

Ответственный: Волков Василий Николаевич

Добавить участников

© 2023 - ИС "Планировщик событий"

Рисунок 50 – Форма создания события

Когда форма заполнена, модератор должен добавить участников события, нажав на кнопку «Добавить участников». Добавление происходит на новой странице, показанной на рисунке 51.

Календарь мероприятий    Добавить пользователя    Запланировать мероприятие    Заявки на регистрацию    Telegram    Выйти

### Добавить участников мероприятия

Боев Сергей Иванович ИТ-специалист ГБУЗ РХ «Черногорская межрайонная больница»  
Андрианов Константин Викторович ИТ-специалист ГБУЗ РХ «Черногорская межрайонная больница»  
Баженова Ксения Серафимовна Главная медицинская сестра ГБУЗ РХ «Черногорская межрайонная больница»  
Волков Василий Николаевич ИТ-специалист ГБУЗ РХ «Черногорская межрайонная больница»  
Котов Давид Викторович Врач-специалист ГБУЗ РХ «Черногорская межрайонная больница»  
Антонов Платон Миронович Бухгалтер ГБУЗ РХ «Черногорская межрайонная больница»

Добавить участников

© 2023 - ИС "Планировщик событий"

Рисунок 51 – Страница добавления участников события

Когда участники назначены, событие появляется на странице «Календарь мероприятий», также пользователям приходит оповещение на почту (рисунок 52) и, если пользователь авторизован в чат-боте, на Telegram (рисунок 53).

**Subject:** ИС "Планировщик событий" Вы были добавлены на новое событие  
**From:** <brady19@ethereal.email>  
**To:** <serj.boev.01@gmail.com>  
**Time:** Today at 19:54  
**Message-ID:** <c2254750-0b7b-11ee-9f0e-4f93729061cf@wildduck.email>

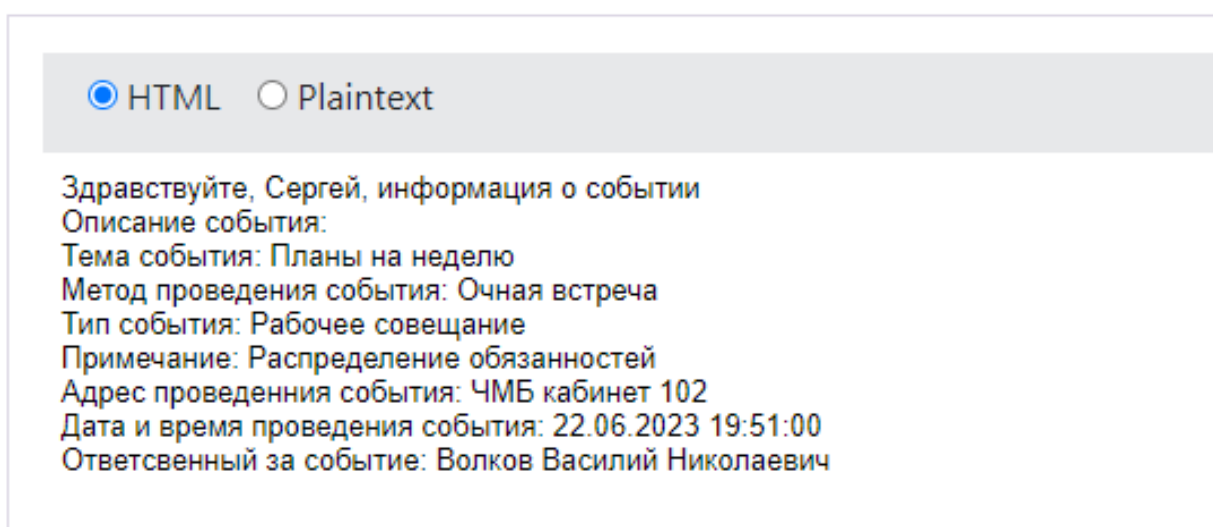


Рисунок 52 – Оповещение о событии через электронную почту

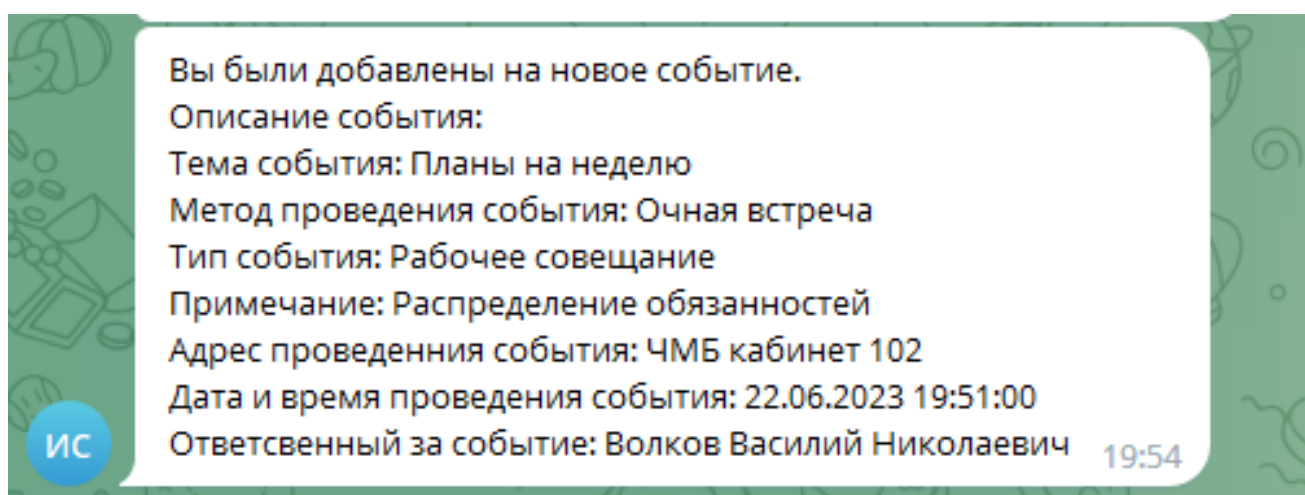


Рисунок 53 – Оповещение о событии через чат-бота Telegram

Также в меню навигации есть пункт «Заявки на регистрацию». Представление данного пункта показано на рисунке 54.

Номер	Фамилия	Имя	Отчество	Организация	Должность	Номер телефона	Электронная почта	Роль	Логин	Кнопка принять	Кнопка отклонить
1	Храпунов	Андрей	Сергеевич	ГБУЗ РХ «Черногорская межрайонная больница»	Заведующий (главный врач, начальник структурного подразделения)	89232109244	qefqef@gmail.com	Пользователь	andrew_off	Принять	Отклонить
2	Михайлов	Павел	Дмитриевич	ГБУЗ РХ «Черногорская межрайонная больница»	Главный врач (начальник, директор)	88005553535	cracazoybra@gmail.com	Модератор	cracazuza	Принять	Отклонить

Рисунок 54 – Представление страницы «Заявки на мероприятия»

На данной странице администратор рассматривает заявки, отправленные пользователями через чат-бота Telegram. Администратор может как отклонить, так и принять заявку. Если заявка принята, то пользователь может пользоваться ИС. Если отклонена, то администратор пишет комментарий, причину отмены, который, в свою очередь, отправляется пользователю через чат-бота, после чего пользователь может отредактировать анкету по указаниям администратора.

Окно для написания комментария представлено на рисунке 55.

Оставьте комментарий по заявке:

Рисунок 55 – Форма написания комментария по заявке

Также, как и у чат-бота все секретные данные, используемые для работы веб-приложения, хранятся в файле конфигурации.

Веб-приложение оснащено системой логирования, которая записывает все действия, происходящие в приложении, в специальный файл с логами (рисунок 56).

```
1 2023-06-15 09:02:23.5965 | DEBUG | Program | init main | url: | action:
2 2023-06-15 09:02:25.7966 | INFO | Microsoft.Hosting.Lifetime | Application started. Press Ctrl+C to shut down. | url: | action:
3 2023-06-15 09:02:25.7966 | INFO | Microsoft.Hosting.Lifetime | Hosting environment: Development | url: | action:
4 2023-06-15 09:02:25.7966 | INFO | Microsoft.Hosting.Lifetime | Content root path: C:\Users\Son-Jin-Woo\source\repos\PlanSob\PlanSob | url: | action:
5 2023-06-15 09:02:25.9311 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/Autohorize | action: Autohorize
6 2023-06-15 09:02:29.4498 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/AutohorizePost | action: AutohorizePost
7 2023-06-15 09:02:30.1429 | | INFO | PlanSob.Controllers.HomeController | Пользователь admin123 был успешно авторизован | url: https://localhost/Home/AutohorizePost | action: AutohorizePost
8 2023-06-15 09:02:30.1429 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
9 2023-06-15 09:02:31.5855 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DayList/5 | action: DayList
10 2023-06-15 09:02:31.5897 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
11 2023-06-15 09:02:32.9542 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DelEvent/76 | action: DelEvent
12 2023-06-15 09:02:33.1816 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 76 | url: https://localhost/Home/DelEvent/76 | action: DelEvent
13 2023-06-15 09:02:33.1816 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
14 2023-06-15 09:02:33.9381 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DayList/16 | action: DayList
15 2023-06-15 09:02:33.9381 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
16 2023-06-15 09:02:34.7948 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DelEvent/77 | action: DelEvent
17 2023-06-15 09:02:34.7948 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 77 | url: https://localhost/Home/DelEvent/77 | action: DelEvent
18 2023-06-15 09:02:34.7948 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
19 2023-06-15 09:02:35.7619 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DayList/15 | action: DayList
20 2023-06-15 09:02:35.7619 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
21 2023-06-15 09:02:36.6594 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DelEvent/81 | action: DelEvent
22 2023-06-15 09:02:36.6594 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 81 | url: https://localhost/Home/DelEvent/81 | action: DelEvent
23 2023-06-15 09:02:36.7883 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
24 2023-06-15 09:02:37.3768 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 75 | url: https://localhost/Home/DelEvent/75 | action: DelEvent
25 2023-06-15 09:02:37.3768 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 75 | url: https://localhost/Home/DelEvent/75 | action: DelEvent
26 2023-06-15 09:02:37.3885 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
27 2023-06-15 09:02:38.1613 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DayList/23 | action: DayList
28 2023-06-15 09:02:38.1613 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
29 2023-06-15 09:02:38.8611 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DelEvent/78 | action: DelEvent
30 2023-06-15 09:02:38.8708 | | INFO | PlanSob.Controllers.HomeController | Пользователь 12 успешно удалил событие 78 | url: https://localhost/Home/DelEvent/78 | action: DelEvent
31 2023-06-15 09:02:38.8708 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
32 2023-06-15 09:02:39.6831 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DayList/18 | action: DayList
33 2023-06-15 09:02:39.6867 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/action: MainPage
34 2023-06-15 09:02:40.5602 | | DEBUG | PlanSob.Controllers.HomeController | MLog injected into HomeController | url: https://localhost/Home/DelEvent/89 | action: DelEvent
35 2023-06-15 09:02:40.6187 | | ERROR | Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddleware | An unhandled exception has occurred while executing the request. System.InvalidOperationException: Sequence contains no elements
36
```

Рисунок 56 – Часть содержимого файла с логами



## **2.4 Выводы по разделу «Описание разработки информационной системы планирования событий Министерства здравоохранения Республики Хакасия»**

В процессе разработки ИС «Планировщик событий» были разработаны веб-приложение и чат-бот Telegram.

Основными функциями веб-приложения являются:

- регистрация;
- авторизация;
- создание событий;
- просмотр, редактирование, удаление событий;
- просмотр и взаимодействие с заявками от чат-бота Telegram;
- оповещение участников события через электронную почту и чат-бота Telegram;
- создание и просмотр логов веб-приложения.

Основными функциями чат-бота Telegram являются:

- авторизация;
- регистрация;
- проверка электронной почты;
- оповещение участников о новых событиях.

Также было установлено необходимое ПО и создана БД, которая состоит из 2 таблиц, 5 справочников и 1 связующей таблицы.

## **3 Определение совокупной стоимости владения информационной системой планирования событий Министерства здравоохранения Республики Хакасия**

Оценка материальных затрат является одной из важнейших проблем при разработке информационных систем, так как данная оценка показывает

предположительный результат проекта в экономическом плане. В качестве метода проведения данной оценки был выбран традиционный метод TCO.

### 3.1 Расчет затрат на реализацию проекта

TCO (Total cost of ownership) – совокупная стоимость владения. Данный метод предполагает количественную оценку на внедрение и сопровождение программного обеспечения.

Показатель совокупной стоимости владения ИС рассчитывается по формуле

$$TCO = DE + IC1 + IC2, \quad (1)$$

где DE (direct expenses) – прямые расходы;

IC1 (indirect costs) – косвенные расходы первой группы;

IC2 – косвенные расходы второй группы.

Прямые расходы рассчитываются по формуле

$$DE = DE1 + DE2 + DE3 + DE4 + DE5 + DE6 + DE7 + DE8, \quad (2)$$

где DE1 – капитальные затраты;

DE2 – расходы на управление ИТ;

DE3 – расходы на техническую поддержку АО и ПО;

DE4 – расходы на разработку прикладного ПО внутренними силами;

DE5 – расходы на аутсорсинг;

DE6 – командировочные расходы;

DE7 – расходы на услуги связи;

DE8 – другие группы расходов.

### 3.2 Капитальные затраты

Капитальные затраты на разработку информационной системы вычисляются по формуле

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оз}}, \quad (3)$$

где  $K_{\text{пр}}$  – затраты на проектирование ИС;

$K_{\text{тс}}$  – затраты на технические средства;

$K_{\text{лс}}$  – затраты на создание линий связи локальных сетей;

$K_{\text{по}}$  – затраты на программные средства;

$K_{\text{ио}}$  – затраты на формирование информационной базы;

$K_{\text{об}}$  – затраты на обучение персонала;

$K_{\text{оз}}$  – затраты на опытную эксплуатацию.

Расчет проектных затрат производится по формуле

$$K_{\text{пр}} = K_{\text{зп}} + K_{\text{свт}} + K_{\text{ипс}} + K_{\text{проч}}, \quad (4)$$

где  $K_{\text{зп}}$  – затраты на заработную плату программиста;

$K_{\text{свт}}$  – затраты на средства вычислительной техники программиста;

$K_{\text{ипс}}$  – затраты на инструментальные программные средства проектирования;

$K_{\text{проч}}$  – прочие затраты на проектирование.

Для расчета  $K_{\text{зп}}$  необходимо вычислить фонд оплаты труда, который состоит из заработной платы программиста и обязательных отчислений на страховые взносы, которые составляют 30,2% от заработной платы. Расчет зарплаты программиста показан в таблице 1.

Оклад программиста в месяц будет считаться равным минимальному размеру оплаты труда (МРОТ), а именно 16242 рубля. Оклад берется за 25 рабочих дней, т.к. программист работает 5 дней в неделю. Для вычисления заработной платы за 5 рабочих дней необходимо уменьшить оклад в 5 раз. Следовательно, заработная плата программиста в неделю равна 3250 рублей, так как программист будет работать 7 недель, значит, в итоге он получит 22750 рублей (без учета северного коэффициента, региональной надбавки, а также без учета НДФЛ).

Расчет заработной платы программиста с учетом северного коэффициента (30%) и региональной надбавки (30%) рассчитан по тарифному способу оплаты.

Оплата за всю проделанную работу с учетом северного коэффициента и региональной надбавки, а также с учетом НДФЛ представлена в таблице 1.

Таблица 1 – Расчет заработной платы программиста

Доходы, руб.		Расходы, руб.	
Заработная плата	22750		
Северный коэффициент	6825		
Районные надбавки	6825		
Все доходы	36400	НДФЛ	4732
Итого на руки	31668		

Итого  $K_{зп} = 36400 * 1,302 = 47393$  рублей.

Затраты на инструментальные программные средства проектирования представлены в таблице 2.

Таблица 2 – Стоимость программного обеспечения

<b>Наименование программного обеспечения</b>	<b>Стоимость, руб.</b>	<b>Срок использования</b>
Visual Studio	3000	2 месяца
Microsoft Windows 10 Professional(64 bit)	4000	1 год
Office 2019	3000	1 год
Sublime	Бесплатно	Бессрочно
PostgreSQL	Бесплатно	Бессрочно

За 1 год использования программных обеспечений возможно создать в среднем 5 программных продуктов, поэтому стоимость всех программных средств, затраты на которые идут в годовую стоимость, необходимо уменьшить в 5 раз. Следовательно, затраты на инструментальные программные средства проектирования равны

$$K_{\text{инс}} = 7000/5 + 3000 = 4400 \text{ рублей.}$$

Чтобы рассчитать затраты на средства вычислительной техники для проектирования, необходимо запланировать средства на амортизацию оборудования для программиста – компьютер с данными техническими комплектующими. Данный компьютер представлен в таблице 3.

Таблица 3 – Стоимость оборудования

<b>Наименование оборудования</b>	<b>Количество элементов</b>	<b>Стоимость единицы, руб.</b>	<b>Стоимость всего, руб.</b>
Ноутбук Acer Aspire 5 A515-45-R8YH	1	49999	49999
Итого	1		49999

Амортизация проекта рассчитывается по следующей формуле

$$A_{\text{пр}} = \frac{A_{\text{год}}}{K_{\text{рдг}}} * K_{\text{дэ}}, \quad (5)$$

где  $A_{\text{пр}}$  – амортизация проекта;

$K_{\text{рдг}}$  – количество рабочих дней в году – 247 дней;

$K_{\text{дэ}}$  – количество дней эксплуатации – 35 дней;

$A_{\text{год}}$  – годовая амортизация.

Годовая амортизация вычисляется по формуле

$$A_{\text{год}} = C_{\text{б}} * N_{\text{ам}}, \quad (6)$$

где  $C_{\text{б}}$  – это балансовая стоимость – 49999 рублей;

$N_{\text{ам}}$  – норма амортизации.

Норма амортизации вычисляется по формуле

$$N_{\text{ам}} = \frac{100\%}{T_{\text{эк}}}, \quad (7)$$

где  $T_{\text{эк}}$  – срок эксплуатации устройства – 3 года.

$N_{\text{ам}} = 0,34$ .  $A_{\text{год}} = 49999 * 0,33 = 16\,500$  рублей.  $A_{\text{пр}} = 16500/247 * 35 = 2338$  рублей.

Затраты на средства вычислительной техники для проектирования составляют 2338 рублей –  $K_{\text{свт}}$ .

Когда затраты на заработную плату программиста, затраты на инструментальные программные средства проектирования и затраты на средства вычислительной техники для проектирования посчитаны, необходимо

вычислить прочие затраты на проектирование, которые составляют 3% от суммы ранее посчитанных затрат.

$$K_{\text{проч}} = (K_{\text{зп}} + K_{\text{ипс}} + K_{\text{свт}}) * 0,03 = (47393 + 4400 + 2338) * 0,03 = 1624 \text{ рублей.}$$

Затраты на проектирование ИС представлены в таблице 4.

Таблица 4 – Затраты на проектирование ИС

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на проектирование ИС	Затраты на заработную плату программиста	47393
	Затраты на ПО	4400
	Затраты на средства вычислительной техники	2338
	Прочие затраты (3% от всех затрат)	1624
Итого		55755

$$K_{\text{пр}} = 55755 \text{ рублей.}$$

Затраты на технические средства управления включают в себя амортизацию компьютера за срок внедрения программы. Стоимость компьютера приведена в таблице 3, срок внедрения равен 2 дня. Затраты на технические средства управления равны амортизации за 2 дня.

$$K_{\text{тс}} = \frac{49999}{247} * 2 = 405 \text{ рублей.}$$

Затраты на линии связи составляют  $K_{\text{лс}} = 0$  рублей.

Затраты на программные средства составляют 0 рублей, так как разрабатываемая информационная система не требует установки дополнительного программного обеспечения, соответственно,  $K_{\text{по}} = 0$  рублей.

Затраты на формирование информационной базы входят в затраты на заработную плату программиста, следовательно -  $K_{но} = 0$  рублей.

На обучение персонала необходимо потратить 3 часа. Часовая зарплата программиста ( $Z_{час}$ ) равна: заработная плата (зп) в неделю, деленная на 5 рабочих дней и на 10 рабочих часов.

$$Z_{час} = \frac{3250}{5 \cdot 10} = 65 \text{ руб./час.}$$

Соответственно затраты на обучение составят

$$K_{об} = 65 \cdot 3 \cdot 1,6 \cdot 1,302 = 407 \text{ рублей.}$$

В затраты на опытную эксплуатацию входит оплата работы программиста по исправлению ошибок или багов проекта, возникших в течение испытательного срока (1 месяц). Программист будет нанят на срок 1 неделю.

$$K_{оэ} = 3250 \cdot 1,6 \cdot 1,302 = 6770 \text{ рублей.}$$

После вычисления всех необходимых затрат, считаются капитальные затраты по формуле (3).

$$K = 55755 + 405 + 407 + 0 + 0 + 0 + 6770 = 63337 \text{ рублей.}$$

Список капитальных затрат представлен в таблице 5.

Таблица 5 – Список капитальных затрат

<b>Затраты</b>	<b>Сумма, руб.</b>
Затраты на заработную плату разработчика	47393
Затраты на программные средства	4400
Затраты на средства вычислительной техники	2338
Прочие затраты на разработку	1624



### Продолжение таблицы 5

Затраты на технические средства управления	405
Затраты на создание линий связи	0
Затраты на программные средства	0
Затраты на формирование информационной базы	0
Затраты на обучение персонала	405
Затраты на опытную эксплуатацию	6770

### 3.3 Расчет эксплуатационных затрат

Эксплуатационные затраты вычисляются по следующей формуле

$$C = C_{\text{зп}} + C_{\text{ао}} + C_{\text{то}} + C_{\text{лс}} + C_{\text{ни}} + C_{\text{проч}}, \quad (8)$$

где  $C_{\text{зп}}$  – зарплата персонала, работающего с ИС;

$C_{\text{ао}}$  – амортизационные отчисления;

$C_{\text{то}}$  – техническое обслуживание;

$C_{\text{лс}}$  – затраты на использование глобальных сетей;

$C_{\text{ни}}$  – затраты на носители информации;

$C_{\text{проч}}$  – прочие затраты.

Для поддержания работы информационной системы необходимо раз в месяц проводить работы по устранению возникших ошибок, дополнять справочники различной информацией (новые события, новые организации, новые должности). Работы будут проводиться полный рабочий день ежемесячно.

$$C_{\text{зп}} = \frac{22750}{7 \cdot 5} * 12 * 1,302 = 10156 \text{ рублей.}$$

В затраты на амортизационные отчисления входит годовая оплата сервера, на котором будет расположена программа. Оплата сервера в месяц составляет 0 рублей, так как Минздрав обладает собственными серверами.

Затраты на техническое обслуживание составляют 0 рублей, так как ИС будет расположена на сервере Минздрава.  $C_{то} = 0$  рублей.

Затраты на использование глобальных сетей 0 рублей, так как затраты на линии связи были учтены ранее.  $C_{лс} = 0$  рублей.

Все данные будут храниться в сети Минздрава Хакасии, соответственно  $C_{ни} = 0$  рублей.

Прочие затраты составляют 3% от перечисленных ранее эксплуатационных затрат.

$$C_{проч} = 10156 * 0,03 = 305 \text{ рублей.}$$

Рассчитаем эксплуатационные затраты по формуле (8).

$$C = 10156 + 305 = 10461 \text{ рубль.}$$

### **3.4 Прямые затраты**

Прямые затраты рассчитываются по формуле (2).

$$DE = 63337 + 10461 + 2214 = 76012 \text{ рубль.}$$

$$DE1 = 63337 \text{ рублей.}$$

$DE2 = 10461$  рубль, расходы на управление составляют эксплуатационные затраты на оплату труда персонала.

$$DE3 = C_{ао} + C_{то} = 0 \text{ рублей.}$$

$DE4 = 0$ , так как система не нуждается в разработке прикладного ПО.

$DE5 = 0$  рублей, так как данной системе не требуются внешние источники.

DE6 = 0 рублей, так как разработка данной ИС не предусматривает командировки.

DE7 = 0 рублей, так как затраты на линии связи были учтены ранее.

DE8 = 0,03\*(63337+10461) =2214 рублей, прочие расходы составляют примерно 3% от всех расходов.

ТСО = 76012 рубля.

Оценка рисков. Данная ИС является достаточно серьезным проектом, имеющим свои довольно весомые риски, которые необходимо учитывать при его создании.

К таким рискам можно отнести:

1. Недостаточное понимание Заказчика и Разработчика.
2. Увеличение планируемых расходов.
3. Несвоевременная сдача проекта ввиду большого объема работы.

Риски представлены в таблице 6.

Таблица 6 – Риски проекта

Риск	Перечень рисков	Уровень влияния	Вероятность	Возможность предотвращения
Риск соответствия	Недостаточность понимания Заказчика и Разработчика	Высокий	Низкая	Консультация Разработчика с Заказчиком на каждом этапе разработки
Операционный риск	Непредвиденные расходы	Средний	Средняя	Оформление дополнительного договора

## Продолжение таблицы 6

Риск, связанный со сроками сдачи проекта	Несвоевременная сдача проекта	Средний	Низкий	Установка сроков сдачи
------------------------------------------	-------------------------------	---------	--------	------------------------

Риск соответствия. Постоянные консультации разработчика с заказчиком о всех выполненных задачах с составлением отчетности.

Операционный риск. Во избежание риска увеличения предполагаемых расходов, необходимо оформить дополнительный денежный договор, в котором будут учитываться все непредвиденные расходы, которые часто возникают при разработке информационной системы.

Риск, связанный со сроками сдачи проекта. Установка заказчиком точной даты сдачи проекта, постоянная связь с заказчиком.

### **3.5 Выводы по разделу «Определение совокупной стоимости владения информационной системой планирования событий Министерства здравоохранения Республики Хакасия»**

Капитальные затраты на создание данной информационной системы равны 63337 рублей, что является средней стоимостью создания систем такого уровня. Эксплуатационные затраты на пользование данной информационной системой равны 10461 рубль.

Совокупная стоимость владения равна 76012 рублей. Данная стоимость показывает, сколько будет стоить система для владельца.

Информационная система была создана с целью оптимизации внутренних процессов. Проект не имеет коммерческого использования, соответственно получение прибыли от проекта не является основной задачей и не рассчитывалась.

## ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы был проведен анализ предметной области деятельности Министерства здравоохранения Республики Хакасия и ГБУЗ РХ «Республиканский медицинский информационно-аналитический центр».

Была разработана концепция IT-проекта, проанализированы аналогичные программные продукты и обоснована необходимость собственной разработки, поскольку программные продукты конкурентов не подходили под особенности процесса.

Для определения всех особенностей проекта были построены модели действий пользователя в системе модели потоков данных.

Модель действий пользователя позволила спроектировать поведение пользователя в системе и учесть все возможные переходы внутри ИС, тем самым предотвратив возникновение ошибок, которые можно было бы допустить на стадии разработки.

Модель потоков данных позволяет проследить, как движется информация внутри системы и как система взаимодействует с внешними сущностями. В данной модели определяются основные внутренние процессы системы и ее внешние сущности, потоки между ними, а также хранилища данных.

Для хранения информации была спроектирована реляционная база данных.

На этапе проектирования был обоснован выбор средств разработки, таких как язык программирования C# и СУБД PostgreSQL.

Также были разработаны веб-приложение ИС «Планировщик событий» и Telegram-бот, взаимодействующий с данной информационной системой.

Капитальные затраты составили 63337 рублей, эксплуатационные затраты 10461 рубль, совокупная стоимость владения равна 76012 рублей. Так как проект создавался с целью оптимизации внутренних процессов и не имеет коммерческого использования, прибыль от проекта не рассчитывалась.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. За честный бизнес всероссийская система проверки контрагентов // [zachestnyibiznes.ru](http://zachestnyibiznes.ru): [сайт]. – URL: [https://zachestnyibiznes.ru/company/ul/1021900532840\\_1901048793\\_GBUZ-RH-RMIAC#:~:text=Государственное%20бюджетное%20учреждение%20здравоохранения%20республики,и%20связанная%20с%20этим%20деятельность%22](https://zachestnyibiznes.ru/company/ul/1021900532840_1901048793_GBUZ-RH-RMIAC#:~:text=Государственное%20бюджетное%20учреждение%20здравоохранения%20республики,и%20связанная%20с%20этим%20деятельность%22) (дата обращения: 24.04.23).
2. О Министерстве здравоохранения Республики Хакасия // Министерство здравоохранения Республики Хакасия: официальный сайт. – URL: <https://mz19.ru/about/o-ministerstve/> (дата обращения: 25.04.23).
3. Постановление правительства Республики Хакасия от 11.06.2009 // Министерство здравоохранения Республики Хакасия: официальный сайт. – URL: <https://mz19.ru/upload/iblock/26c/Postanovlenie-Pravitelstva-Respubliki-Khakasiya-ot-11.06.2009.pdf> (дата обращения: 26.04.23).
4. Устав Министерства здравоохранения Республики Хакасия // Министерство здравоохранения Республики Хакасия: официальный сайт. – URL: <https://aib.mz19.ru/upload/doc/MIAC/ustav.pdf> (дата обращения: 28.04.23).
5. О веб-приложении // [ru.wikipedia.org](http://ru.wikipedia.org): [сайт]. – URL: <https://ru.wikipedia.org/wiki/Веб-приложение> (дата обращения: 29.04.23).
6. О языке программирования Python // [ru.wikipedia.org](http://ru.wikipedia.org): [сайт]. – URL: <https://ru.wikipedia.org/wiki/Python> (дата обращения: 01.05.23).
7. Интернет технологии. ru // [www.internet-technologies.ru](http://www.internet-technologies.ru): [сайт]. – URL: <https://www.internet-technologies.ru/articles/php-ruby-python-harakteristika-yazykov-programmirovaniya.html> (дата обращения: 01.05.23).
8. О языке программирования C# // [ru.wikipedia.org](http://ru.wikipedia.org): [сайт]. – URL: [https://ru.wikipedia.org/wiki/C\\_Sharp](https://ru.wikipedia.org/wiki/C_Sharp) (дата обращения: 02.05.23).
9. DVE // [drach.pro](http://drach.pro): [сайт]. – URL: <https://drach.pro/blog/hi-tech/item/145-db-comparison> (дата обращения: 03.05.23).
10. НСЕ-реестр справочников // [nsi.rosminzdrav.ru](http://nsi.rosminzdrav.ru): [сайт]. – URL:

<https://nsi.rosminzdrav.ru/dictionaries/1.2.643.5.1.13.13.11.1002/passport/7.7> (дата обращения: 05.05.23).

11. Медицинские организации, подведомственные Министерству здравоохранения Республики Хакасия // [mz19.ru](http://mz19.ru): [сайт]. – URL: <https://mz19.ru/clinic/> (дата обращения: 07.05.23).

Выпускная квалификационная работа выполнена мной самостоятельно.  
Использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

Отпечатано в одном экземпляре.

Библиография 11 наименований.

Один экземпляр сдан на кафедру.

« \_\_\_\_ » \_\_\_\_\_ 2023 г.


\_\_\_\_\_ Боев Сергей Иванович  
подпись



Министерство науки и высшего образования РФ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Хакасский технический институт – филиал ФГАОУ ВО  
«Сибирский федеральный университет»


Кафедра прикладной информатики, естественно-научных  
и гуманитарных дисциплин


УТВЕРЖДАЮ  
Заведующий кафедрой  
 О. В. Папина  
подпись  
« 19 » июня 2023 г.

**БАКАЛАВРСКАЯ РАБОТА**

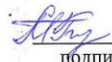
09.03.03 Прикладная информатика

Разработка информационной системы планирования событий  
Министерства здравоохранения Республики Хакасия

Руководитель  19.06.23 ст. преподаватель В. И. Кокова  
подпись, дата

Выпускник  19.06.2023 г. С. И. Боев  
подпись, дата

Консультанты  
по разделам:

Экономический  19.06.2023 г. М. А. Бурева  
подпись, дата

Нормоконтролер  19.06.2023 А. Н. Кадычегова  
подпись, дата

Абакан 2023