

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, математики и естественно-научных  
дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Е. Н. Скуратенко  
                                подпись  
«\_\_\_\_\_» \_\_\_\_\_ 2022 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.03 Прикладная информатика

Разработка чат-бота «Расписание МБДОУ «Рябинка»

Руководитель \_\_\_\_\_ доцент, канд. физ.-мат. наук А. Н. Таскин  
                                подпись, дата

Выпускник \_\_\_\_\_ Т. Е. Галицына  
                                подпись, дата

Консультанты  
по разделам:

Экономический \_\_\_\_\_ Е. Н. Скуратенко  
                                подпись, дата

Нормоконтролер \_\_\_\_\_ В. И. Кокова  
                                подпись, дата

Абакан 2022

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, математики и естественно-научных  
дисциплин

УТВЕРЖДАЮ  
Заведующий кафедрой  
\_\_\_\_\_ Е. Н. Скуратенко  
подпись  
«\_\_\_\_\_» \_\_\_\_\_ 2022 г.

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

Абакан 2022

Студенту Галицыной Татьяне Евгеньевной

Группа ХБ 18-03

Направление 09.03.03 Прикладная информатика

Тема выпускной квалификационной работы: Разработка чат-бота «Расписание МБДОУ «Рябинка».

Утверждена приказом по институту № 208 от 14.04.2022 г.

Руководитель ВКР: А. Н. Таскин, доцент, канд. физ.-мат. наук, ХТИ – филиал СФУ

Исходные данные для ВКР: заказ МБДОУ «Рябинка».

Перечень разделов ВКР:

1. Анализ процесса создания чат-бота «Расписание МБДОУ «Рябинка»
2. Описание разработки чат-бота «Расписание МБДОУ «Рябинка».
3. Расчёт затрат и оценка экономической эффективности чат-бота «Расписание МБДОУ «Рябинка».

Перечень графического материала: нет

Руководитель ВКР

\_\_\_\_\_

подпись

А. Н. Таскин

Задание принял к исполнению

\_\_\_\_\_

подпись

Т. Е. Галицына

«14» апреля 2022 г.

## РЕФЕРАТ

Выпускная квалифицированная работа по теме «Разработка чат-бота «Расписание МБДОУ «Рябинка» содержит 65 страниц текстового документа, 38 иллюстраций, 11 формул, 11 таблиц, 21 использованный источник, приложение А.

ПРОГРАММНЫЙ ПРОДУКТ, ПЛАНИРОВАНИЕ, РАЗРАБОТКА, ЧАТ-БОТ, REG.RU, PUSCHARM, PYTHON, SQLITE, ЭКСПЛУАТАЦИОННЫЕ ЗАТРАТЫ, КАПИТАЛЬНЫЕ ЗАТРАТЫ, РИСКИ ПРОЕКТА, ЭКОНОМИЧЕСКАЯ ЭФФЕКТИВНОСТЬ, РАСПИСАНИЕ.

Цель выпускной квалифицированной работы: разработка чат-бота «Расписание МБДОУ «Рябинка».

В ходе выполнения работы были выполнены следующие задания:

В первом разделе проанализирована деятельность МБДОУ «Рябинка»; обоснована необходимость разработки; выбрана среда разработки.

Во втором разделе спроектирован и разработан чат-бот «Расписание МБДОУ «Рябинка».

В третьем разделе произведены расчеты затрат и оценка экономической эффективности проекта.

Результатом работы является спроектированный и разработанный чат-бот «Расписание МБДОУ «Рябинка», позволяющий сократить временные затраты на поиски необходимого расписания.

## SUMMARY

The theme of the graduation thesis is «Chat-bot Development: Schedule of Classes in Municipal Budgetary Pre-school Educational Institution “Ryabinka”» (MBPEI). It contains 65 pages, 38 figures, 11 formulae, 11 tables, 21 reference items.

SOFTWARE PRODUCT, PLANNING, DEVELOPMENT, CHATBOT, REG.RU, PYCHARM, PYTHON, SQLITE, OPERATING COSTS, CAPITAL COSTS, PROJECT RISKS, ECONOMIC EFFICIENCY, SCHEDULE.

The purpose of the thesis: the development of a chat-bot «Schedule of Classes in MBPEI “Ryabinka”».

Objectives:

The first section analyzes the performance area of MBPEI “Ryabinka”; the need for development has been suggested; the development environment has been selected.

The second section describes the way the chat-bot «Schedule of Classes in MBPEI “Ryabinka”» has been designed and developed.

The third section shows costs calculations and assessment of economic efficiency of the project.

The chat-bot «Schedule of Classes in MBPEI “Ryabinka”» has been designed and developed; it allows to reduce the time spent on searching for necessary schedule.

English language supervisor: \_\_\_\_\_

N.V. Chezybaeva

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	8
1 Анализ процесса создания чат-бота «Расписание МБДОУ «Рябинка».....	10
1.1 Характеристика деятельности МБДОУ «Рябинка» .....	10
1.2 Актуальность предметной области.....	12
1.3 Описание бизнес-процесса просмотра необходимой информации AS-IS .	14
1.4 Описание основных функций разрабатываемого чат-бота .....	16
1.5 Обоснование и выбор средств разработки чат-бота «Расписание МБДОУ «Рябинка».....	18
1.5.1 Выбор архитектуры ПП чат-бот «Расписание МБДОУ «Рябинка» .....	18
1.5.2 Выбор средств проектных решений .....	20
1.5.2.1 Выбор платформы.....	20
1.5.2.2 Выбор сервера .....	21
1.5.2.3 Выбор языка программирования .....	23
1.5.2.4 Выбор среды разработки .....	25
1.5.2.5 Выбор системы управления базами данных.....	27
1.6 Вывод по разделу «Анализ процесса создания чат-бота «Расписание МБДОУ «Рябинка».....	29
2 Описание разработки чат-бота «Расписание МБДОУ «Рябинка».....	29
2.1 Описание бизнес-процесса получения необходимой информации ТО-ВЕ .	30
2.2 Диаграмма потоков данных чат-бота «Расписание МБДОУ «Рябинка»....	33
2.3 Пошаговая разработка чат-бота «Расписание МБДОУ «Рябинка» .....	35
2.3.1 Регистрация чат-бота .....	35
2.3.2 Разработка чат-бота .....	38
2.4 Выводы по разделу «Разработка чат бота «Расписание МБДОУ «Рябинка» .....	54
3 Расчет затрат и оценка экономической эффективности чат-бота «Расписание МБДОУ «Рябинка» .....	54
3.1 Расчет затрат на разработку чат-бота «Расписание МБДОУ «Рябинка» ...	54
3.2 Капитальные затраты .....	55
3.3 Расчет эксплуатационных затрат .....	62
3.4 Расчет совокупной стоимости владения системой .....	65
3.5 Экономическая эффективность реализации проекта.....	66
3.6 Оценка риска при реализации проекта создания программного продукта	66

3.7 Выводы по разделу «Расчет затрат и оценка экономической эффективности чат-бота «Расписание МБДОУ «Рябинка» .....	68
ЗАКЛЮЧЕНИЕ .....	69
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	70
ПРИЛОЖЕНИЕ А .....	72

## ВВЕДЕНИЕ

В настоящее время основой всего стали информационные технологии, а в частности интернет. Сейчас уже трудно представить какую-либо сферу человеческой деятельности без использования компьютера или базы данных. Все завязано на использовании вычислительной техники и трудно, уже представить нашу жизнь без всего этого. Общение тоже в свою очередь перешло из реального мира в виртуальный, мы каждый день пользуемся социальными сетями, мессенджерами и другими приложениями для общения с людьми. Это стало важным компонентом повседневной жизни.

Актуальность темы выпускной квалифицированной работы обусловлена большой популярностью мессенджеров и в частности чат-ботов. Чат-бот – специальный алгоритм, формирующий переписку с настоящим собеседником. Чат-боты помогают и упрощают жизнь, выполняя однотипные задачи.

Целью выпускной квалифицированной работы является разработка чат-бота «Расписание МБДОУ «Рябинка».

Для достижения цели необходимо решить следующие задачи:

1. ознакомиться с деятельностью МБДОУ «Рябинка»;
2. выполнить анализ бизнес-процесса просмотра необходимой информации;
3. выполнить описание процесса просмотра необходимой информации средствами моделирования;
4. определить основные характеристики системы;
5. обосновать выбор средств разработки программного продукта;
6. построить функциональную и информационную модели ПП;
7. разработать и описать чат-бот «Расписание МБДОУ «Рябинка»;
8. рассчитать затраты и оценить экономическую эффективность разработки чат-бота «Расписание МБДОУ «Рябинка»;
9. оценить риски реализации проекта;
10. оформить текст выпускной квалифицированной работы.



При выполнении выпускной квалифицированной работы использовались методы исследования:

- теоретический – поиск, систематизация, анализ информации, моделирование;
- эмпирический – выбор инструментов, средств разработки, разработка программного продукта и её функциональное тестирование.

Теоретический результат выпускной квалифицированной работы представлен в первом разделе.

Эмпирический результат представлен во втором и в третьем разделах. Во втором разделе представлено описание разработки чат-бота «Расписание МБДОУ «Рябинка» и в третьем разделе расчет затрат, описание оценки экономической эффективности и рисков реализации разработанного программного продукта.

# **1 Анализ процесса создания чат-бота «Расписание МБДОУ «Рябинка»**

## **1.1 Характеристика деятельности МБДОУ «Рябинка»**

Муниципальное бюджетное дошкольное образовательное учреждение Детский сад «Рябинка», сокращенное наименование МБДОУ «Рябинка».

Учредителем Детского сада является публично-правовое образование – муниципальное образование город Черногоorsk в лице Администрации г. Черскогогорска.

Функции и полномочия Учредителя осуществляет городское управление образованием администрации города Черногогорска (ГУО).

Образовательная программа дошкольного образования Муниципального бюджетного дошкольного образовательного учреждения детского сада «Рябинка» утверждена решением педагогического совета (протокол № 1) от 26.08.2021, приказ № 127 и определяет содержание и организацию образовательной деятельности в Учреждении.

Адаптированная образовательная программа дошкольного образования Муниципального бюджетного дошкольного образовательного учреждения детского сада «Рябинка» утверждена решением педагогического совета (протокол № 1) от 26.08.2021, приказ № 127 и характеризует систему организации образовательной деятельности с детьми групп компенсирующей направленности детей с тяжелыми нарушениями речи и обеспечивает построение целостного образовательного процесса.

В детском саду функционируют 4 группы: 3-общеразвивающей направленности и 1-группа компенсирующей направленности, для детей с ограниченными возможностями здоровья (тяжелое нарушение речи и фонетико-фонематическое нарушение недоразвитие речи)

Детский сад рассчитан на 90 детей от 1,5 до 7 лет. Комплектуется в мае каждого года. Прием детей осуществляется на основании медицинского

заклучения, заявления и документов, удостоверяющих личность одного из родителей (законных представителей). Группа детей с ограниченными возможностями здоровья комплектуется на основании заключения психолого-медико-педагогической комиссии.

МБДОУ «Рябинка» является звеном муниципальной системы образования, обеспечивающее помощь семье в воспитании детей дошкольного возраста, охрану и укрепления их физического и психического здоровья, развитию индивидуальных способностей и необходимую коррекцию нарушений развития этих детей. Воспитатели проводят с детьми разные занятия, ходят на прогулки и детские мероприятия. Для того, чтобы грамотно распределить время, воспитатели составляют расписания, оно помогает не теряться в занятиях и не забывать про запланированные мероприятия.

На рисунке 1 изображена структурная схема учреждения. В МБДОУ «Рябинка» имеется администрация, основной педагогический состав, старшая мед. сестра и младший обслуживающий персонал.

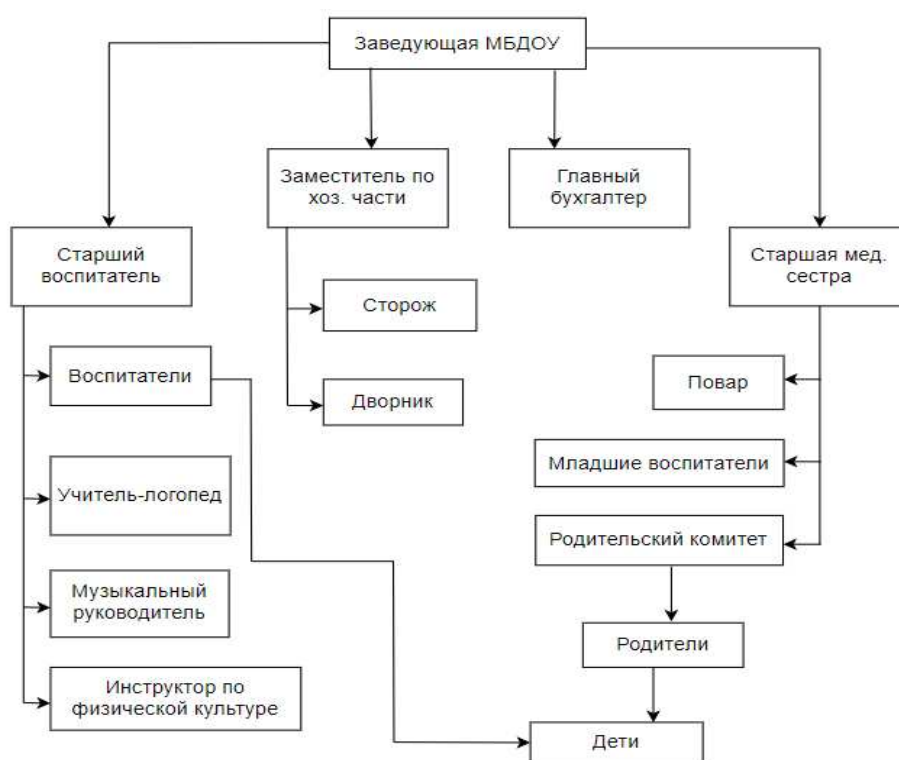


Рисунок 1 – Структурная схема МБДОУ «Рябинка»

Администрация состоит из: заведующей, старшего воспитателя, заместителя заведующей по хозяйственной части. В ее задачи входит: руководство детским садом, организация всего воспитательно-образовательного процесса и работа по хозяйственному обслуживанию.

В основной педагогический состав входят: воспитатель, учитель-логопед, музыкальный руководитель и инструктор по физической культуре. Основной функцией является обеспечение образовательного процесса.

Старшая мед. сестра занимается медицинскими вопросами детского сада. К младшему обслуживающему персоналу относятся: младшие воспитатели, повар, сторож, дворник. Все эти сотрудники обеспечивают бесперебойную работу детского сада, но только младшие воспитатели работают непосредственно с детьми.

## **1.2 Актуальность предметной области**

Расписание МБДОУ «Рябинка» составляют воспитатели каждой своей группы, далее оно утверждается у старшего воспитателя. Каждый воспитатель пишет и составляет рабочую программу на год для своей группы, эту программу размещают на официальном сайте МБДОУ «Рябинка» города Черногорск. Данные программы содержат расписание занятий каждой группы, которое могут посмотреть любой желающий.

Актуальность выпускной квалифицированной работы состоит в том, что расписание, расположенное на сайте долго находить, особенно если не знаешь где искать, на это занятие придется затратить неопределенное количество времени. Чат-бот поможет быстро предоставить необходимое расписание и затратить меньшее количество времени на поиски, а также предоставить информацию по имеющимся кружкам в данном детском саду.

Для того, чтобы найти расписание на сайте, необходимо зайти на официальный сайт и в меню сайта выбрать «Образование». На рисунке 2 представлена часть меню сайта.

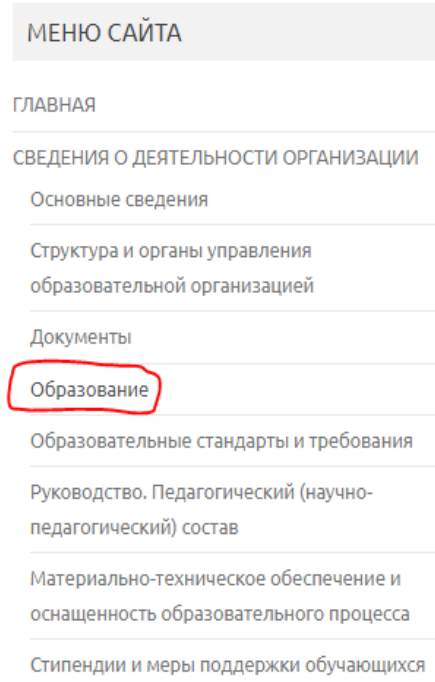


Рисунок 2 – Выбор кнопки "Образование" на панели меню

Далее выводится страница «Образование», пролистав страницу вниз, находим рабочую программу для групп общеразвивающей направленности. Затем выбирается интересующая группа по годам. На рисунке 3 представлены документы рабочей программы на сайте.

#### **[Рабочая программа младшей группы общеразвивающей направленности 3-4 лет 2021-2022](#)**

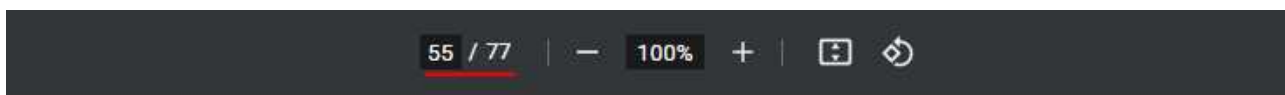
Рабочая программа является составным компонентом Образовательной программы дошкольного образования МБДОУ «Рябинка». Содержание программы выстроено с учетом специфики дошкольного возраста. Реализация программы способствует развитию эстетических чувств детей, познавательной активности, интереса к жизни своей страны, республики, города, воспитывает культуру общения, стремление к активной деятельности и творчеству. В программе представлено календарно-тематическое планирование образовательной работы с детьми по всем образовательным областям. При завершении реализации Программы проводится диагностика, связанная с оценкой эффективности педагогических действий с целью дальнейшей оптимизации и индивидуализации. Диагностика проводится два раза в год (сентябрь, май). Срок реализации программы 1 год

#### **[Аннотация к Рабочей программе педагога разновозрастной группы общеразвивающей направленности 3- 4 лет](#)**

#### **[Рабочая программа средней группы общеразвивающей направленности 4-5 лет 2021 -2022 уч.г.](#)**

Рисунок 3 – Рабочая программа на сайте

Пройдя по гиперссылке скачивается и открывается PDF-документ, для поиска расписания, необходимо пролистать на большое количество страниц вниз. Это довольно время затратное дело. На рисунке 4 представлено количество страниц, которое необходимо пролистать, чтобы найти расписание.



## ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ

### Распорядок / режим дня

Распределение режимных моментов в течение дня в соответствии с возрастом в холодный период (сентябрь - апрель)

Рисунок 4 – Пролистываемое количество страниц

Такое количество действий для получения расписания утомляет, особенно если постоянно повторять, если расписание обновляется.

### 1.3 Описание бизнес-процесса просмотра необходимой информации AS-IS

Диаграмма *AS-IS* (как есть), описывающая поиск необходимой информации, представлена на рисунке 5.

Центральный блок – поиск необходимой информации:

- вход: официальный сайт МБДОУ «Рябинка»;
- механизмы: пользователь (родители, педагоги и др. сотрудники);
- управление: правила сайта;
- выход: расписание занятий (распорядок дня).

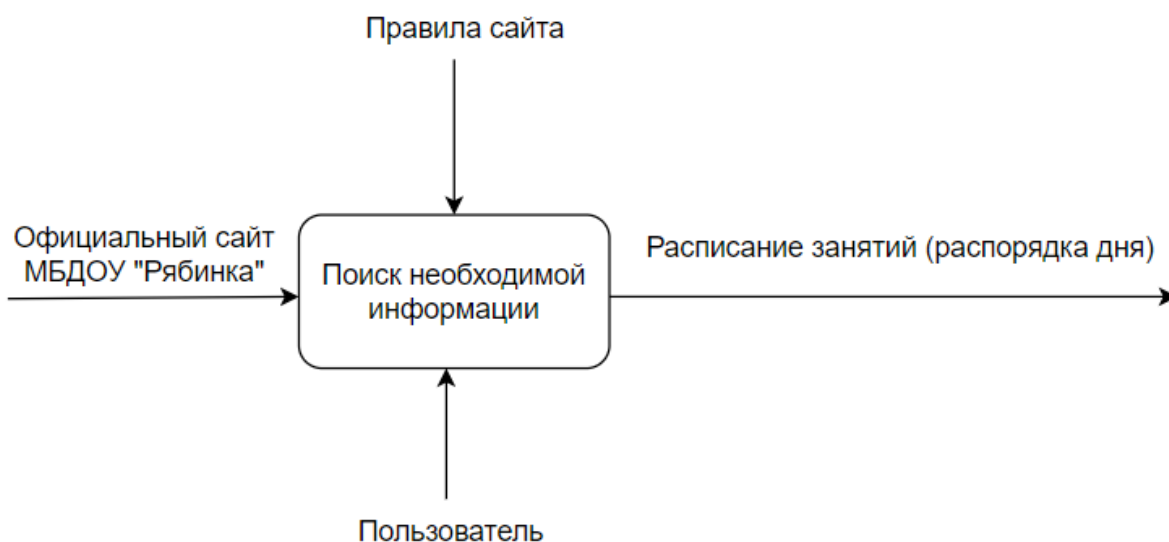


Рисунок 5 – Диаграмма AS-IS «Поиск необходимой информации»

Блок «Поиск необходимой информации» был декомпозирован на 4 блока, что отображено на рисунке 6.

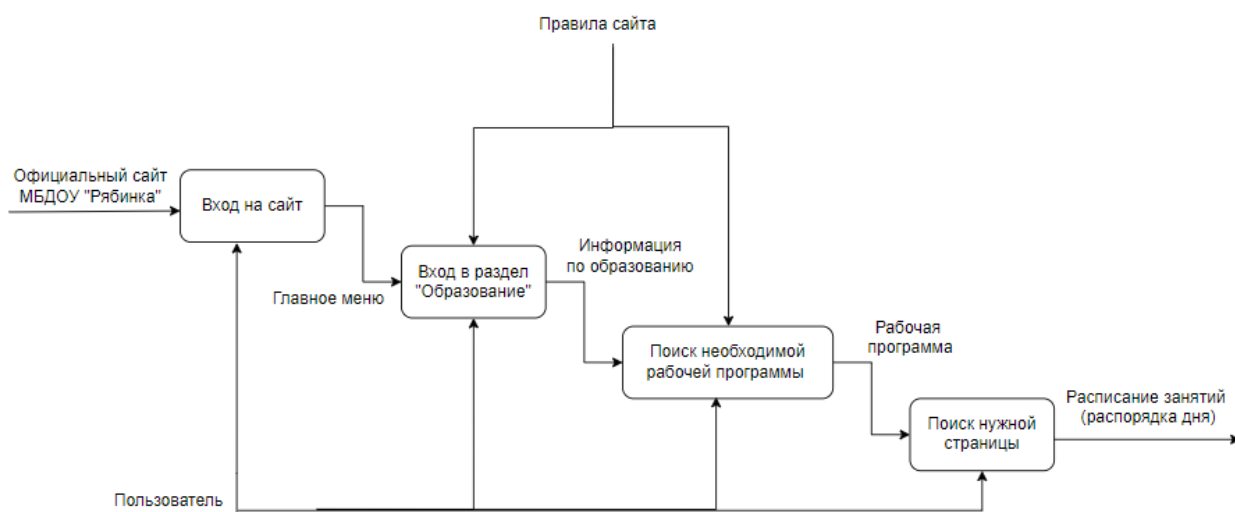


Рисунок 6 – Декомпозиция блока «Поиск необходимой информации»

Блок «Вход на сайт» – необходимо зайти на официальный сайт МБДОУ «Рябинка», где находится необходимая информация.

Блок «Вход в раздел Образование» – необходимо выбрать в меню сайта раздел «Образование».

Блок «Поиск необходимой рабочей программы» – нужно найти рабочую программу интересующей группы.

Блок «Поиск нужной страницы» – открыв интересующую рабочую программу (PDF-документ), пользователем осуществляется поиск страницы, на которой расположено расписание занятий (распорядок дня).

#### **1.4 Описание основных функций разрабатываемого чат-бота**

Для сокращения времени и упрощения поиска необходимого расписания или информации по кружкам в МБДОУ «Рябинка» было принято решения разработать чат-бот расписания.

Разрабатываемый программный продукт должен быть доступен любому пользователю Telegram.

Расписание, подготовленное воспитателями каждой группы и утвержденное у старшего воспитателя, должно вводиться в чат-бот администратором.

Для получения информации по имеющимся кружкам, пользователь должен выбрать интересующий его кружок и бот предоставит по нем информацию.

Для того, чтобы увидеть необходимое расписание, пользователь для начала должен выбрать интересующую его группу. Затем он должен решить, хочет он посмотреть все расписание, расписание на сегодня, завтра или только на определенный день. После сделанного выбора, например определенного дня, выбирается день недели, и чат-бот предоставляет расписание.

Администратор в свою очередь может, как добавлять информацию по кружкам и расписанию, так ее и изменять.

Для того, что наглядно отобразить функциональные возможности чат-бота построена диаграмма UseCase (рисунок 7).

Use Case Diagram – диаграмма вариантов использования, отражающая отношения между актерами и прецедентами и являющаяся составной частью



модели прецедентов, позволяющей описать систему на концептуальном уровне [8].

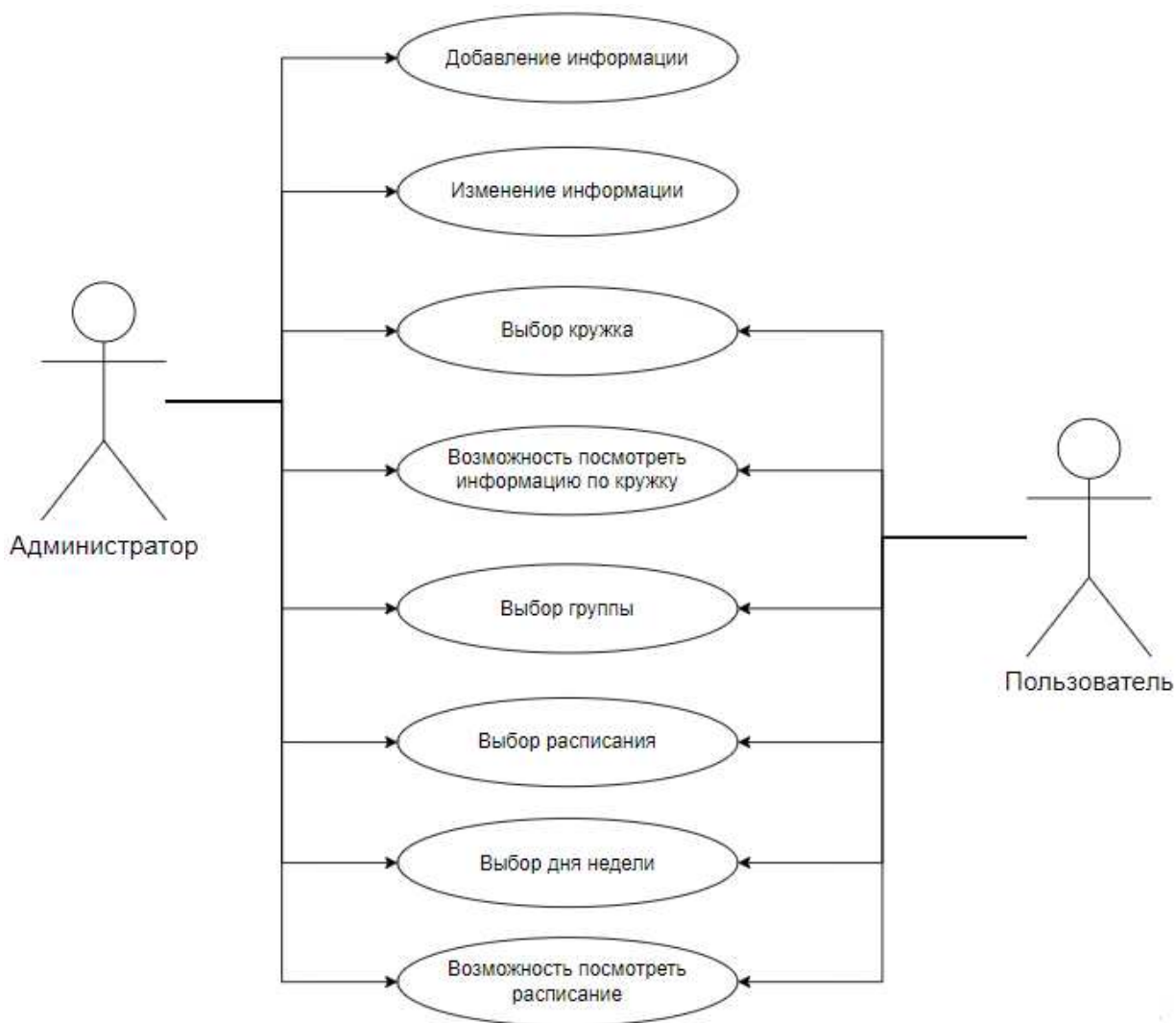


Рисунок 7 – Диаграмма UseCase

Разрабатываемый программный продукт чат-бот «Расписание МБДОУ «Рябинка» должен предоставлять:

- список кружков, имеющих в МБДОУ «Рябинка»;
- информацию по кружкам;
- список групп, находящиеся в МБДОУ «Рябинка»;
- занятия по времени;
- мероприятия, по времени их появления.

Целью разработки и внедрения программного продукта чат-бот «Расписание МБДОУ «Рябинка» является быстрое предоставление необходимого расписания и информации по кружкам сотрудникам детского сада и родителям.

Для реализации разработки необходимо решить следующие задачи:

- ознакомиться с расписанием занятий;
- ознакомиться с информацией по кружкам;
- выбрать инструментальные средства разработки;
- создать проект.

## **1.5 Обоснование и выбор средств разработки чат-бота «Расписание МБДОУ «Рябинка»**

### **1.5.1 Выбор архитектуры ПП чат-бот «Расписание МБДОУ «Рябинка»**

Большинство обычных чат-ботов не являются интеллектуальными приложениями. Они используют для выдачи ответа лишь заранее заложенные шаблоны, не адаптируются сами под текущие потребности пользователей, редко автоматически выполняют аналитику, не являются обучаемыми, не реализуют автоматическое пополнение базы знаний, или вовсе не содержат базы знаний, а ограничены просто базой данных. Всё что они могут, это определить намерение в сообщении пользователя и вернуть ему наиболее релевантный из имеющихся в базе ответов.

Чат-бот должен быть построен так, чтобы быть способным взаимодействовать с популярными мессенджерами, такими как Telegram, WhatsApp, Viber. Это повышает внедряемость, а следовательно, и потенциальную полезность, и востребованность такого приложения. Многие мессенджеры предоставляют готовый веб-сервис, который обеспечивает работу

с API мессенджера. Из этого следует важный вывод, что перед нами не стоит задачи построения пользовательского интерфейса чат-бота и веб-сервиса обеспечивающего работу приложения с этим интерфейсом. Эти задачи высококачественно решены и реализованы популярными сервисами обмена сообщениями ввиду понимания ими важности и актуальности применения чат-ботов. Существует большое количество чат-ботов, работающих по такому принципу.

На рисунке 8 схематически представлена типичная архитектура простого чат-бота.



Рисунок 8 – Архитектура чат-бота

## 1.5.2 Выбор средств проектных решений

### 1.5.2.1 Выбор платформы

Для того, чтобы начать разрабатывать чат-бот, необходимо определиться с платформой размещения. Составим таблицу 1 популярных мессенджеров и выберем подходящую платформу.

Таблица 1 – Сравнительная таблица популярных мессенджеров

	<b>Telegram</b>	<b>WhatsApp</b>	<b>Viber</b>
Плюсы	<ul style="list-style-type: none"><li>– добавление “умных” ботов в чаты;</li><li>– секретный чат;</li><li>– возможность создавать групповые чаты до 200 000 пользователей;</li><li>– веселые анимированные стикеры, можно создать свои уникальные стикеры;</li><li>– открытый исходный код;</li><li>– интегрированный редактор фотографий; протокол шифрования MTProto.</li></ul>	<ul style="list-style-type: none"><li>– протокол HTTPS;</li><li>– есть возможность отправки оффлайн-сообщений;</li><li>– анализ данных пользователя;</li><li>– возможность создавать групповые чаты до 256 пользователей;</li><li>– можно совершать голосовые вызовы.</li></ul>	<ul style="list-style-type: none"><li>– большой архив эмоджи и стикеров;</li><li>– уникальная игровая платформа;</li><li>– функция групповых чатов чатов;</li><li>– можно совершать аудио и видео звонки;</li><li>– уведомления работают даже в оффлайн-режиме;</li><li>– определяет местоположение юзера.</li></ul>
Минусы	<ul style="list-style-type: none"><li>– на десктопной версии нет секретных чатов.</li></ul>	<ul style="list-style-type: none"><li>– нет игровой платформы;</li><li>– качество передаваемых медиафайлов заметно искажается.</li></ul>	<ul style="list-style-type: none"><li>– проблемы с безопасностью;</li><li>– очень популярный мессенджер для спама и рекламы.</li></ul>

После сравнения популярных мессенджеров и проведения беседы с педагогами и вышестоящим руководством МБДОУ «Рябинка» для размещения чат-бота было принято решение использовать мессенджер Telegram. Кроме этого основной мессенджер для коммуникации между родителями и воспитателями является Telegram.

Telegram по праву считается лучшей площадкой для общения, а сегодня он и самый шумевший, и все благодаря его создателю. Это не первый проект Павла Дурова, который так пришелся по душе пользователю. А глядя на то, какая война ведется за сохранение секретности данных пользователя и то, что для безопасности используется криптографический протокол MTProto, это несомненные плюсы в пользу мессенджера. Telegram — является абсолютно бесплатным приложением, благодаря ему пользователи могут обмениваться текстовыми, аудио и видео сообщениями. Изначально это было только мобильным приложением для iOS и Android, однако позже появилась ПК версия. Юзеры особенно оценят функцию «секретный чат», который можно увидеть, только на том устройстве, где он был создан, на всех остальных, даже на синхронизированных устройствах, его видно не будет [2] .

### **1.5.2.2 Выбор сервера**

Хостинг для сайта — это услуга предоставления мощностей для размещения информации о сайте на специальном веб-сервере. Такое размещение нужно для того, чтобы посетители всегда могли зайти на ресурс и найти его в интернете.

Виртуальный хостинг (shared hosting) — это один из типов хостинга, при котором множество веб-сайтов расположено на одном веб-сервере. Часто понятия «хостинг» и «виртуальный хостинг» используют как синонимы.

VPS/VDS — одна из разновидностей хостинга, услуга предоставления в аренду виртуального выделенного сервера. Особенностью этого типа является то, что клиенту достается виртуальный аналог обычного физического сервера со всеми вытекающими последствиями.

При аренде VPS предоставляется root-доступ, который позволяет устанавливать любое программное обеспечение, необходимое для работы сайта и любые необходимые библиотеки и приложения.

VPS получает собственный ip-адрес, которым не придется делиться с соседями. Это важно, так как часто бывает, что, под бан гугла попадает не конкретный сайт, а вся группа сайтов, работавших через один ip-адрес. Кроме того, персональный ip повышает противозломную защиту данных.

Работая на VPS, можно гибко и оперативно перераспределять наличные ресурсы между своими проектами. Разумеется, также нет ограничений в количестве сайтов, которые размещаются на сервере, количестве баз данных или почтовых ящиков.

Также стоит отметить, что на VPS можно создать сервер чат-бота на любом языке программирования.

Для реализации программного продукта необходимо выбрать регистратор доменных имен, на котором будет совершена покупка хостинг-провайдера для загрузки разрабатываемого чат-бота. Для этого проведем сравнение между тремя самыми популярными российскими регистраторами доменных имен. Составим таблицу 2, где будут представлены плюсы и минусы каждого регистратора.

Таблица 2 – Сравнительная таблица российских регистраторов доменных имен

	<b>Reg.ru</b>	<b>Мастерхост</b>	<b>Макхост</b>
Плюсы	<ul style="list-style-type: none"> <li>– выгодные цены на VPS;</li> <li>– моментальная активация виртуального сервера;</li> <li>– неограниченный трафик;</li> <li>– хороший саппорт;</li> <li>– SSD-диски;</li> <li>– промокоды Google AdWords;</li> <li>– бесплатный ISPmanager на месяц;</li> <li>– бесплатные SSLсертификаты;</li> <li>– бесплатные домены в зонах .RU/.РФ.</li> </ul>	<ul style="list-style-type: none"> <li>– Hyper-V VPS;</li> <li>– бесплатный трафик на VPS;</li> <li>– бесплатный трафик при соблюдении.</li> </ul>	<ul style="list-style-type: none"> <li>– круглосуточная поддержка;</li> <li>– в стоимость VPS входит помощь в переносе сайтов от других хостинг-компаний;</li> <li>– онлайн поддержка с отличными специалистами;</li> <li>– бесплатный домен ru при оплате хостинга сразу за год.</li> </ul>
Минусы	<ul style="list-style-type: none"> <li>– нет тестового периода ;</li> <li>– тех поддержка долго не отвечает.</li> </ul>	<ul style="list-style-type: none"> <li>– дорого;</li> <li>– VPS медленные;</li> <li>– техподдержка отвечает слишком медленно.</li> </ul>	<ul style="list-style-type: none"> <li>– дорогой;</li> <li>– нет windows VPS;</li> <li>– VPS отказываются переносить партнерскую программу.</li> </ul>

Сравнив плюсы и минусы трех выбранных российских регистраторов доменных имен, выбор был сделан в пользу регистратора Reg.ru. У него больше плюсов и меньше минусов.

Reg.ru — российский регистратор доменных имён и хостинг-провайдер, аккредитованный регистратор в доменах .ru и .рф. На июль 2017 года компания обслуживала около 3 миллионов доменов и предоставляла хостинг для около 400 тысяч доменных имён. С момента основания Reg.ru специализируется на двух основных направлениях: регистрации доменов (на 2017 год — более 750 доменных зон) и предоставлении хостинга. Компания предоставляет инструменты для создания, продвижения и защиты сайтов. Также Reg.ru продаёт сертификаты удостоверяющих центров Comodo, Symantec, GeoTrust , GlobalSign [20].

### 1.5.2.3 Выбор языка программирования

Для того, чтобы разработать чат-бот необходимо выбрать язык программирования. Составив таблицу 3 сравнения нескольких языков, выбирается самый подходящий.

Таблица 3 – Сравнительная таблица языков программирования

	<b>Python</b>	<b>PHP</b>	<b>JavaScript</b>
Плюсы	<ul style="list-style-type: none"> <li>– Низкий порог вхождения. Синтаксис Python более понятный для новичка.</li> <li>– Логичный, лаконичный и понятный. В сравнении с многими другими языками Python имеет легко читаемый синтаксис, разве что Visual Basic тоже легок.</li> <li>–</li> </ul>	<ul style="list-style-type: none"> <li>– Кроссплатформенность. PHP может быть запущен в любой операционной системе, включая юниксоиды.</li> <li>– Поддержка веб-серверов. Сложно найти тот, который бы не работал с PHP.</li> <li>– Бесплатное распространение.</li> <li>–</li> </ul>	<ul style="list-style-type: none"> <li>– Быстрый для конечного пользователя: сценарий Java написан для клиентской стороны, для поддержки веб-сервера, не требуется поддержка.</li> <li>– Простота: JavaScript относительно прост в освоении и реализации.</li> <li>–</li> </ul>

Продолжение таблицы 3

	<ul style="list-style-type: none"> <li>– Кроссплатформенный: подходит для разных платформ: и Linux, и Windows.</li> <li>– Python отличается строгим требованием к написанию кода (требует отступы), что является преимуществом, по моим наблюдениям. Изначально язык способствует писать код организованно и красиво.</li> </ul>	<ul style="list-style-type: none"> <li>– Имеет достаточную произвольность для web-разработки.</li> </ul>	<ul style="list-style-type: none"> <li>– Универсальность: JavaScript отлично работает с другими языками и может использоваться в самых разных приложениях</li> </ul>
Минусы	<ul style="list-style-type: none"> <li>– Низкая скорость выполнения программ, по сравнению с другими языками.</li> <li>– Копирование кода. При копировании кода с другого ресурса, в некоторых случаях, он может скопироваться без сохранения отступов. Поэтому код будет невалидным, а Вам придется долго добавлять табуляцию в каждую строчку.</li> </ul>	<ul style="list-style-type: none"> <li>– Узкопрофильность.</li> <li>– Безопасность. У PHP есть средства безопасности уровня системы и уровня web-приложения. Но, опять же, широкая используемость сыграла злую шутку: дыры в PHP находят быстрее, чем разработчики успевают их закрывать. В PHP 7 множество проблем решено, но злоумышленник всегда впереди. В силу того, что массы знают «препроцессор», трудно предугадать всё.</li> <li>– Противоречия в коде. Когда шаблонизатор был только создан, все программное обеспечение разрабатывались с помощью C. Потому в языке было применено множество синтаксиса из него. В то же время, современная аудитория больше сконцентрирована на Java. В итоге, код переполнен различными остатками из разных языков. И все они могут даже быть сконцентрированы в одном выражении кода.</li> </ul>	<ul style="list-style-type: none"> <li>– Безопасность: JavaScript явно добавлен к веб-страницам и клиентским браузерам, он может использовать систему пользователя, поэтому вредоносный код может быть запущен на клиентской машине.</li> <li>– Поддержка браузера: JavaScript иногда интерпретируется по-разному разными браузерами. Различные механизмы компоновки могут отображать JavaScript по-разному, что приводит к несогласованности с точки зрения функциональности и интерфейса.</li> <li>– Отключить JavaScript: если вы отключите JavaScript в браузере, весь код JavaScript не запущен.</li> <li>– Загрузка файла: файл JavaScript загружается на клиентской машине, чтобы каждый мог прочитать код и повторно использовать его.</li> </ul>



Сравнив языки программирования, выбор был сделан в пользу языка Python. Он прост в использовании и легок для понимания. У него большое комьюнити с хорошей поддержкой.

Python является высокоуровневым языком программирования, который был разработан в 1991 году Гвидо ван Россумом. Этот язык предлагает динамическое управление типами данных и поддерживает различные парадигмы программирования, включая объектно-ориентированную, процедурную или функциональную. В 2018 году популярность Python возросла, он стал одним из самых популярных языков. Практически во всех рейтингах он входит в тройку лучших языков программирования.

Python был разработан как проект с открытым кодом, который предлагает бесплатные установочные пакеты для всех популярных платформ, таких как Unix, Windows, macOS, Android.

На данный момент существует 3 версии Python, из которых поддерживается только последняя третья версия. Версия Python 3.0 была выпущена в 2008 году.

#### 1.5.2.4 Выбор среды разработки

Выбирая среду разработки, стоит обратить внимания на выбранный язык программирования. Редактор кода должен поддерживать язык программирования Python. Составим таблицу 4 сравнения редакторов кода.

Таблица 4 – Сравнительная таблица редактора кода

	Sublime Text	Visual Studio Code	PyCharm
Описание	свободное программное обеспечение с широкой поддержкой сообщества. Редактор работает с несколькими языками программирования. В большинстве случаев можно	бесплатный редактор кода от Microsoft для Windows, Linux и MacOS. Его возможности — отладка, подсветка синтаксиса, интеллектуальное завершение кода, предопределённые фрагменты	это интегрированная среда разработки специально для Python. Разработана чешской [прим. перев. — автор ориентируется на расположение головного офиса] компанией JetBrains.

Продолжение таблицы 4

	использовать незарегистрированный Sublime Text, но время от времени вы будете получать всплывающее окно с просьбой зарегистрировать продукт и приобрести лицензию. Sublime Text тонко настраивается и дополняется пакетами для отладки, автозавершения кода, линтинга и т.д. [3].	кода, рефакторинг и интеграция с Git. Поддерживаются различные языки программирования. Для начала работы с Python может понадобиться несколько дополнительных пакетов, но установить их довольно просто. Редактор постоянно обновляется. Visual Studio Code — один из лучших редакторов не только для Python, но и для других языков программирования [3].	Редактор разработан специально для Python, так что имеет широкий набор возможностей, таких как автозавершение и инспекции кода, подсветка ошибок, исправления, отладка, система контроля версий и рефакторинг. IDE доступна на Microsoft Windows, Linux и MacOS. Есть бесплатная и платная профессиональная версии. Профессиональная IDE имеет несколько дополнительных функций, но бесплатной версии достаточно для большинства задач [3].
Плюсы	<ul style="list-style-type: none"> <li>– Простой и по большей части бесплатный.</li> <li>– Тонко настраивается.</li> <li>– Компактный и эффективный.</li> </ul>	<ul style="list-style-type: none"> <li>– Это платформа с непрерывными обновлениями.</li> <li>– Потребляет немного памяти по сравнению с другими громоздкими инструментами разработки.</li> <li>– Имеет встроенный терминал и прост в использовании.</li> </ul>	<ul style="list-style-type: none"> <li>– Имеет важные встроенные функции.</li> <li>– Разработана профессионалами специально для Python.</li> <li>– Поддерживает виртуальные среды Anaconda</li> </ul>
Минусы	Для удобства требует дополнительных пакетов	Иногда терминал работает не так, как хотелось бы.	Если на ПК или ноутбуке не 8 Гб оперативной памяти, то IDE немного притормаживает и работает довольно медленно.

Выбор между редакторами кода был сделан в пользу PyCharm, он разработан специально для языка Python и не требует больших знаний для того, чтобы разобраться, как с ним работать.

### 1.5.2.5 Выбор системы управления базами данных

#### **MySQL.**

MySQL — это реляционная система управления базами данных с открытым исходным кодом. В настоящее время эта СУБД одна из наиболее популярных в веб-приложениях — подавляющее большинство CMS использует именно MySQL, а почти все веб-фреймворки поддерживают MySQL уже на уровне базовой конфигурации (без дополнительных модулей).

Из преимуществ СУБД MySQL стоит отметить простоту использования, гибкость, низкую стоимость владения (относительно платных СУБД), а также масштабируемость и производительность.

MySQL позволяет хранить целочисленные значения со знаком и беззнаковые, длиной в 1, 2, 3, 4 и 8 байтов, работает со строковыми и текстовыми данными фиксированной и переменной длины, позволяет осуществлять SQL-команды SELECT, DELETE, INSERT, REPLACE и UPDATE, обеспечивает полную поддержку операторов и функций в SELECT- и WHERE- частях запросов, работает с GROUP BY и ORDER BY, поддерживает групповые функции COUNT(), AVG(), STD(), SUM(), MAX() и MIN(), позволяет использовать JOIN в запросах, в т.ч. LEFT OUTER JOIN и RIGHT OUTER JOIN, поддерживает репликацию, транзакции, работу с внешними ключами и каскадные изменения на их основе, а также обеспечивает многие другие функциональные возможности.

#### **MongoDB.**

MongoDB — это документоориентированная система управления базами данных, которая не требует описания схемы таблиц. Считается одним из классических примеров NoSQL-систем, использует JSON-подобные документы и схему базы данных. Написана на языке C++.

В реляционной базе данных информация представлена как набор связанных между собой таблиц. Доступ к данным осуществляется с помощью запросов на языке SQL. Реляционная модель — традиционная, ее используют чаще всего. Но для многих задач она не оптимальна, например, если связи между разными данными не жесткие.

В MongoDB реализована система, при которой данные хранятся в «табличных» документах форматов, близких к JSON. Информация записывается в виде пар из ключей и значений — как в таблице, в которой есть идентификаторы и соответствующие им данные. Благодаря такому формату в MongoDB можно разместить очень разнообразную и сложно структурированную информацию: документ — более гибкая структура.

При работе с такими базами SQL не используется, отсюда название NoSQL. Вместо него применяют языки программирования. В случае с MongoDB это JavaScript. Существуют драйверы для поддержки других популярных языков: Python, Java, C/C++, Go, PHP, Ruby и прочих.

### **SQLite.**

SQLite — это система управления базами данных, отличительной особенностью которой является ее встраиваемость в приложения. Это значит, что большинство СУБД являются самостоятельными приложениями, взаимодействие с которыми организовано по принципу клиент-сервер. Программа-клиент посылает запрос на языке SQL, СУБД, которая в том числе может находиться на удаленном компьютере, возвращает результат запроса.

В свою очередь SQLite является написанной на языке с библиотекой, которую динамически или статически подключают к программе. Для большинства языков программирования есть свои привязки (API) для библиотеки SQLite. Так в Python СУБД SQLite импортируют командой `import sqlite3`. Причем модуль `sqlite3` входит в стандартную библиотеку языка и не требует отдельной установки.

Приложение, включающее в себя SQLite, использует ее функциональность посредством простых вызовов функций. Поскольку функции вызываются в том же процессе, что работает приложение, вызовы работают быстрее, чем это было бы в случае межпроцессного взаимодействия.

Просмотрев несколько существующих систем управления базами данных, выбор был сделан в пользу SQLite. SQLite отлично работает в качестве механизма базы данных для большинства веб-сайтов с низким и средним трафиком (то есть для большинства веб-сайтов). Объем веб-трафика, который может обрабатывать SQLite, зависит от того, насколько интенсивно веб-сайт использует свою базу данных.

## **1.6 Вывод по разделу «Анализ процесса создания чат-бота «Расписание МБДОУ «Рябинка»»**

Проанализировав деятельность МБДОУ «Рябинка» была выявлена актуальность создания чат-бота, для сокращения временных затрат на поиск необходимого расписания и информации по кружкам, имеющимся в детском саду «Рябинка» для сотрудников учреждения и родителей.

Был смоделирован и описан бизнес-процесс поиска необходимой информации AS-IS (как есть).

После сравнения и консультации с сотрудниками МБДОУ «Рябинка» был выбран мессенджер Telegram для размещения разрабатываемого чат-бота.

Для реализации программного продукта были выбраны средства разработки с помощью сравнения: Reg.ru, Python, PyCharm, SQLite.

## **2 Описание разработки чат-бота «Расписание МБДОУ «Рябинка»»**

Разработка чат-бота «Расписание МБДОУ «Рябинка» осуществляется поэтапно в соответствии с планом. Основная цель планирования разработки –

обеспечить исполнение работ и достижение конечных результатов проекта. В таблице 5 представлен поэтапный план работ по разработке чат-бота «Расписание МБДОУ «Рябинка» с кратким описанием каждого этапа.

Таблица 5 – План работ по разработке проекта

№ п/п	Наименование этапа	Описание этапа
1.	Подготовка серверного ПО	Подключение необходимых модулей для работы системы
2.	Регистрация чат-бота	Регистрация бота в системе и получение токена для привязки чат-бота.
3.	Создание меню	Написание программного кода для кнопок чат-бота.
4.	Подключение администратора	Написание программного кода для входа в бот как администратор.
5.	Создание дополнительных функций администратору	Написание программного кода для дополнительных функций администратору

## 2.1 Описание бизнес-процесса получения необходимой информации ТО-ВЕ

Описание функций пользователя и администратора при наличии механизма ПП чат-бота «Расписание МБДОУ «Рябинка» происходят с помощью диаграмм.

Центральный блок – «Получение необходимой информации» (рис. 9).

Вход: вход в чат-бот.

Управление: требования заказчика.

Механизмы: пользователь, администратор, Telegram.

Выходы: расписание занятий (распорядок дня), информация по кружкам.

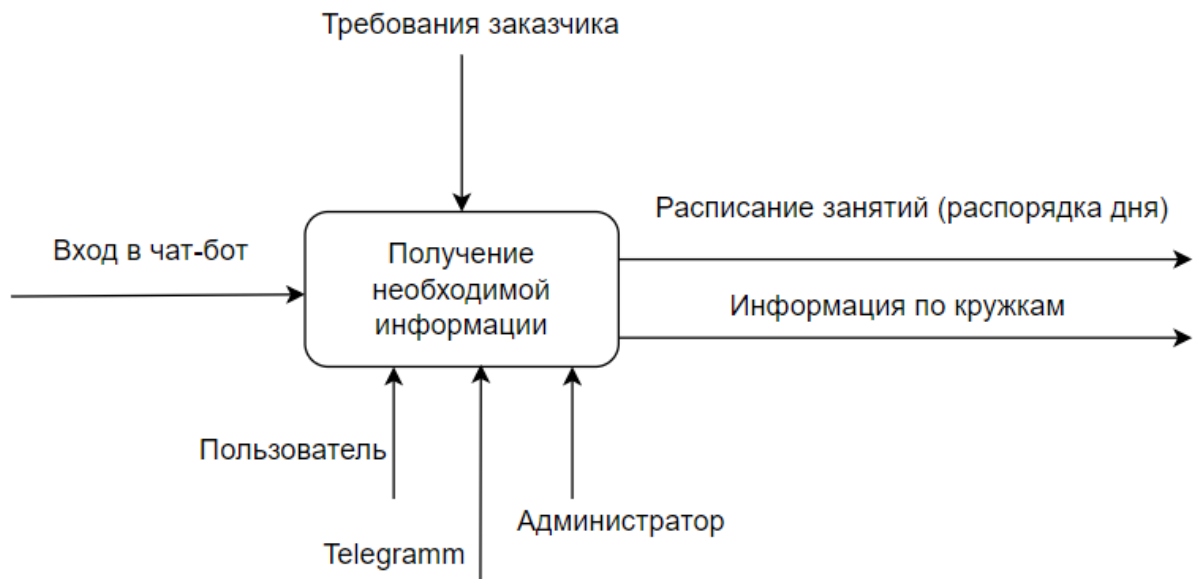


Рисунок 9 – Диаграмма ТО-ВЕ «Получение необходимой информации»

На рисунке 10 изображена декомпозиция диаграммы ТО-ВЕ .

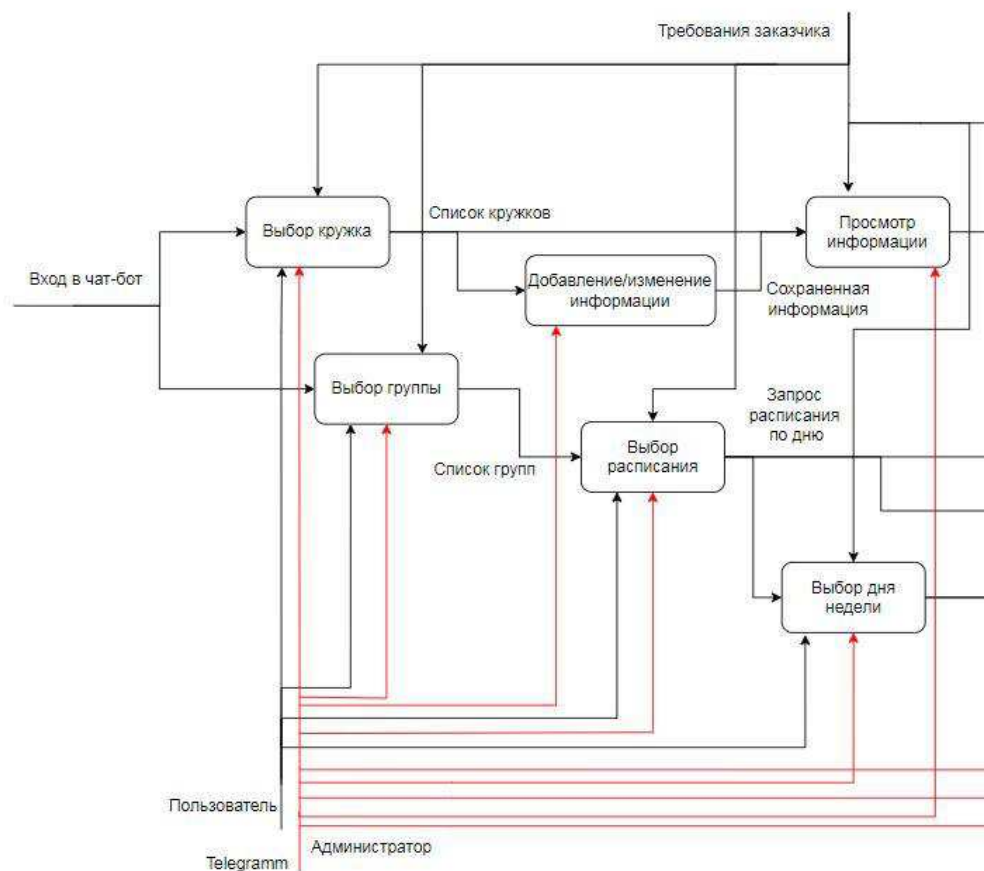


Рисунок 10 – Декомпозиция диаграммы ТО-ВЕ, лист 1

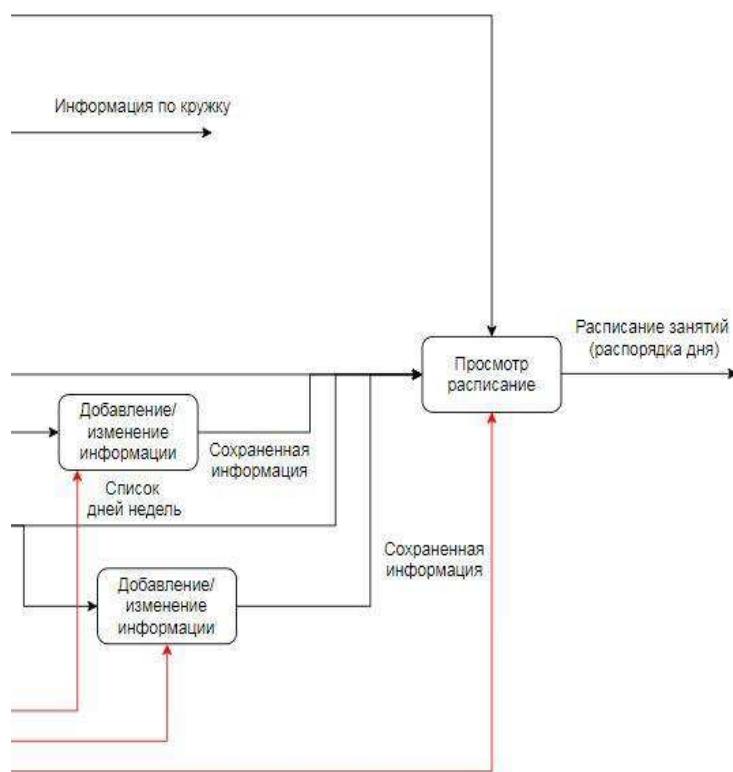


Рисунок 11, лист 2

Блок «Выбор кружка» – необходимо выбрать кружок из выпадающего списка, информацию по которому необходимо посмотреть. Выходом из этого блока является «Список кружков», «Информация по кружку», механизмами «Пользователь», «администратор», «Telegram».

Блок «Выбор группы» – необходимо выбрать группу из выпадающего списка, расписание которой будет выводиться. Выходом из данного блока является «Список групп», механизмами «Пользователь», «администратор», «Telegram».

Блок «Выбор из списка» – бот предоставляет список того, какое расписание необходимо вывести, в списке есть: «Сегодня», «Завтра», «День недели», «Все расписание».

Если выбрать «Сегодня», «Завтра», «Все расписание», то дальнейших выборов не будет и бот выведет выбранное расписание. Выходом является



«Запрос расписания по дню», «Расписание занятий (распорядок дня)», механизмами «Пользователь», «администратор», «Telegram».

Блок «Выбор дня недели» – необходимо выбрать день недели с понедельника по пятницу. Выбрав день, бот выведет расписание на этот день. Выходом из блока является «Список дней недели», «Расписание занятий (распорядок дня)», механизмами «Пользователь», «администратор», «Telegram».

Блоки «Добавление/изменение информации» – доступ к данным кнопкам есть только у администратора, он может, как добавить информацию, так ее и изменить.

Требования заказчика формируют внешний вид чат-бота и ее функционал.

Разработанный программный продукт позволит быстро найти необходимое расписание любой группы, а также предоставит информацию по имеющимся кружка в МБДОУ «Рябинка», сократив время на поиски печатного вида.

## **2.2 Диаграмма потоков данных чат-бота «Расписание МБДОУ «Рябинка»**

Диаграмма потоков данных (DFD). DFD — общепринятое сокращение от англ. data flow diagrams — диаграммы потоков данных. Так называется методология графического структурного анализа, описывающая внешние по отношению к системе источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ. Модель системы в нотации DFD представляет собой совокупность иерархически упорядоченных и взаимосвязанных диаграмм.

На рисунке 11 отображена диаграмма потоков данных программного продукта «Чат-бот «Расписание МБДОУ «Рябинка»».

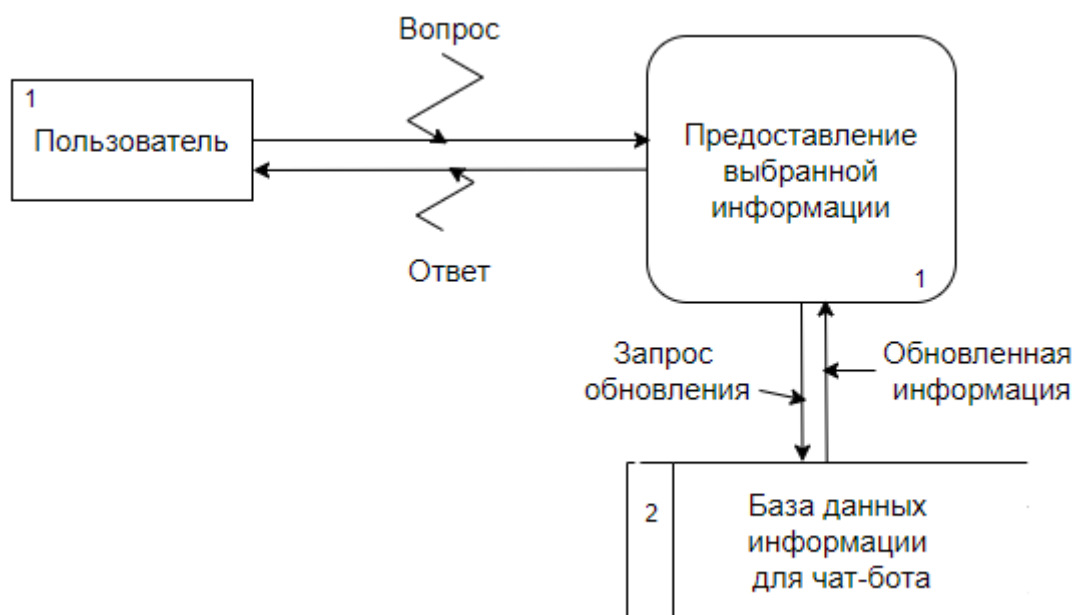


Рисунок 12 – Диаграмма потоков данных

Разберем диаграмму, которая изображена на рисунке 11. Процесс 1 является функциональным, он принимает данные и отправляет их. Помимо функционального блока присутствует внешняя сущность и накопитель данных. Пользователь, внешняя сущность, для того чтобы получить нужную ему информацию, он выбирает информацию, которая ему необходима на панели, а в ответ ему передается требуемая информация. Блок «База данных информации для чат-бота» – накопитель данных, после получения запроса на обновление информации, в базе данных изменяется данная информация и отправляется уже обновленная в чат-бот.

Проведем декомпозицию контекстной диаграммы, чтобы лучше разобрать, как осуществляется поток данных внутри системы. Декомпозиция изображена на рисунке 12.

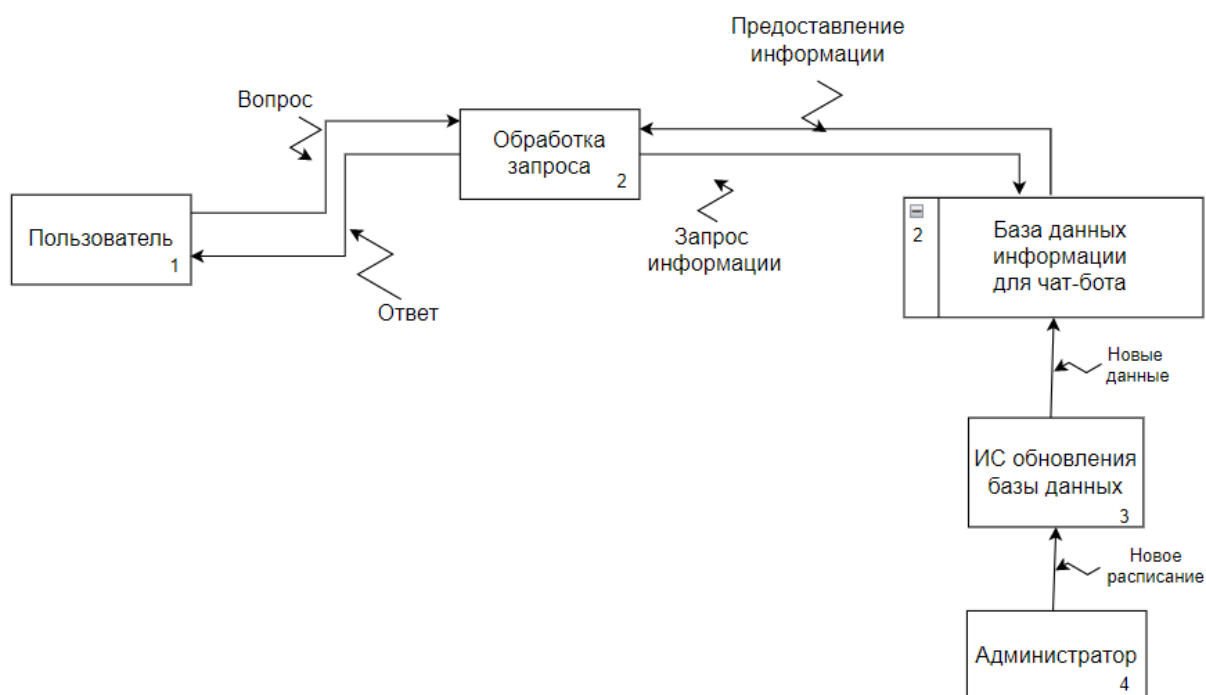


Рисунок 12 – Декомпозиция DFD-диаграммы

Разберем декомпозицию на рисунке 12. Внешняя сущность 1 «Пользователь», задает вопрос внутренней системе 2 «Обработка запроса», она же в свою очередь делает запрос необходимой информации в блок «База данных информации для чат-бота», тот предоставляет необходимую информацию и в конце «Пользователь» получает ответ, через систему 2. Если информация изменилась, то внешняя сущность 4 «Администратор» вносит эти изменения во внутреннюю систему 3 «ИС обновления базы данных», она же в свою очередь изменяет необходимую информацию в блоке «База данных информации для чат-бота».

## 2.3 Пошаговая разработка чат-бота «Расписание МБДОУ «Рябинка»

### 2.3.1 Регистрация чат-бота

Самым первым шагом разработки чат-бота мессенджера Telegram является регистрация бота в системе и получения токена для привязки чат-бота.

Регистрация проходит в месенджере. Для создания нового бота необходимо отправить команду «/newbot» чат-боту «BotFather». Этот бот создан специально для регистрации и управления всеми чат-ботами Telegram. На рисунке 13 показан процесс регистрации чат-бота.

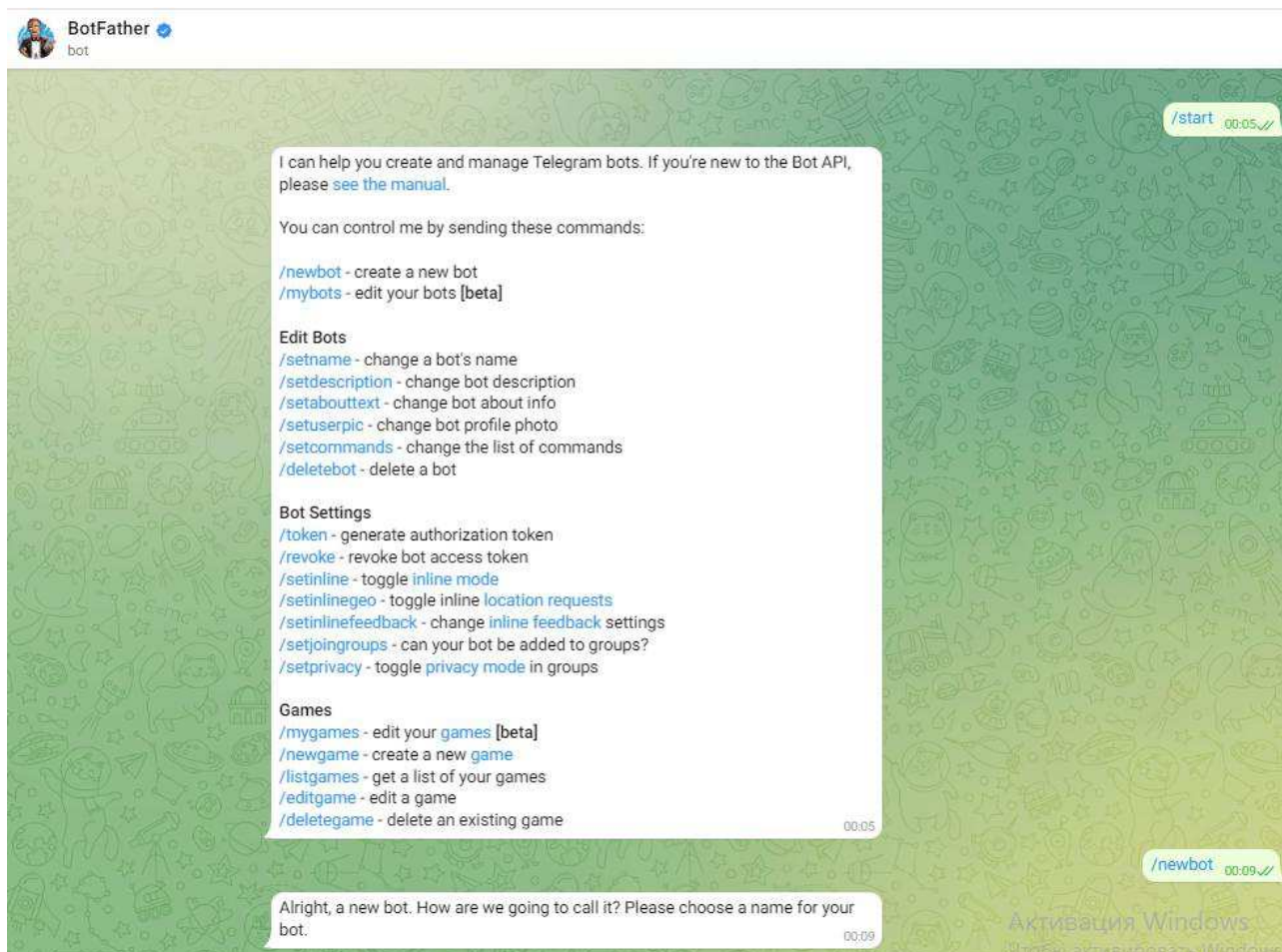


Рисунок 13 – Регистрация чат-бота

Далее вводится имя будущего бота: «Расписание занятий МБДОУ «Рябинка». Затем отправляется адрес бота, который должен заканчиваться на слово «\_bot»: “sad\_rabinka\_bot”. После успешной регистрации «BotFather» выдает токен чат-бота. Токен – это уникальный ключ доступа к создаваемому чат-боту. Данный токен указывается в коде создаваемого чат-бота. На рисунке 14 показано присвоение имени, адреса и получения токена.

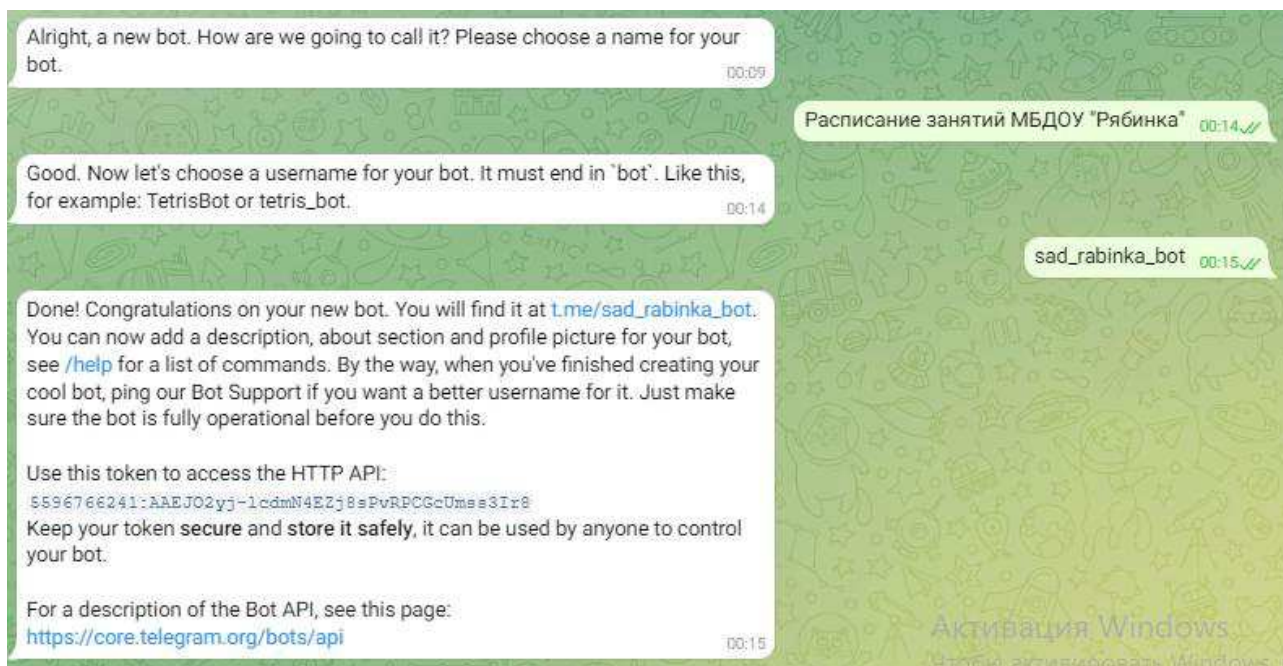


Рисунок 14 – Присвоение имени и получение токена

Кроме того, для настройки чат-ботов в «BotFather» в будущем можно использовать следующие команды:

- /setname – изменение имени;
- /setdescription – изменение описания бота;
- /setabouttext – изменение информации о боте;
- /setuserpic – изменение фотографии о боте;
- /setcommands – изменение списка команд бота;
- /setjoiningroups – изменение настроек добавления бота в группы;
- /setprivacy – изменение настроек видимости сообщений в группах;
- /deletebot – удаления бота.

Кроме команд изменения настроек чат-бота, существуют команды, позволяющие изменять параметры и присваивать нужные значения:

- /token – отправка полученного ранее токена;
- /revoke – аннулирования токена доступа к боту;
- /setinline – настройка возможности вызова бота из других чатов;
- /setinlinegeo – настройка возможности передачи местоположения.

Данные настройки можно менять в любое время. Все параметры изменяются моментально у всех пользователей чат-бота.

После регистрации чат-бота можно приступить к разработке программного кода.

### **2.3.2 Разработка чат-бота**

При разработке чат-бота использовались python-библиотеки aiogram, environs. Aiogram – это простой и полностью асинхронный фреймворк для Telegram Bot API. Асинхронность позволяет отвлекаться от задач и не ждать ввода от пользователя, поэтому используются библиотека aiogram.

Environ – среда пользователя. Среда пользователя представляет собой совокупность информации о пользователе, такой как его входной каталог, электронный почтовый ящик, тип терминала. Среда хранится в специальных «переменных среды», которым могут присваиваться значения символов, таких как имена файлов, каталогов и терминалов. Эти переменные автоматически имеют доступ к программам и командам, вызываемым пользователем и терминалом.

Для удобства пользования чат-ботом используется меню [Приложение 1] в виде кнопок. Такое меню позволяет пользователю видеть список действий, на которые чат-бот сможет ответить.

В самом начале пользователю предоставляется главное меню, которое содержит две кнопки:

- кружки;
- группы.

На рисунке 15 представлено главное меню.

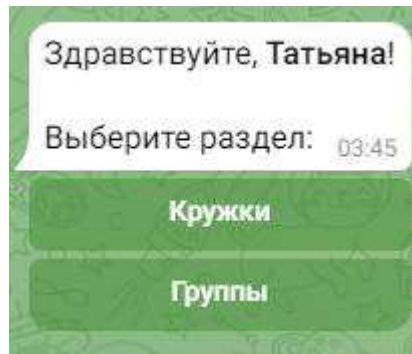


Рисунок 15 – Главное меню

Данная клавиатура (меню) реализуется с помощью объекта `InlineKeyboardMarkup`. Каждая клавиатура, созданная таким образом, привязана к сообщению чат-бота, с которым была отправлена. Программный код реализации главного меню, представлен на рисунке 16.

```
from aiogram import types
from aiogram.dispatcher.filters import Text
from aiogram.dispatcher.filters.builtin import CommandStart

from loader import dp
from data import config

@dp.message_handler(CommandStart())
@dp.callback_query_handler(Text('start'))
async def start(message: types.Message):
    '''Запуск бота'''
    msg_text = (
        f'Здравствуйте, <b>{message.from_user.first_name}</b>!\n\n'
        f'Выберите раздел:'
    )
    keyboard = types.InlineKeyboardMarkup(row_width=1)
    keyboard.add(
        types.InlineKeyboardButton('Кружки', callback_data='lessons'),
        types.InlineKeyboardButton('Группы', callback_data='groups')
    )

    try:
        await message.answer(msg_text, reply_markup=keyboard)
    except:
        await message.message.edit_text(msg_text, reply_markup=keyboard)
```

Рисунок 16 – Код главного меню

Нажав на кнопку «Кружки», пользователю предоставляется список кружков в виде кнопок, которые имеются в детском саду «Рябинка». Данный список состоит из четырех кружков:

- музыка;
- рисование;
- развивайка;
- рукоделие.

Для возврата в предыдущий раздел, необходимо нажать на кнопку «Назад». На рисунке 17 продемонстрировано меню имеющихся кружков.

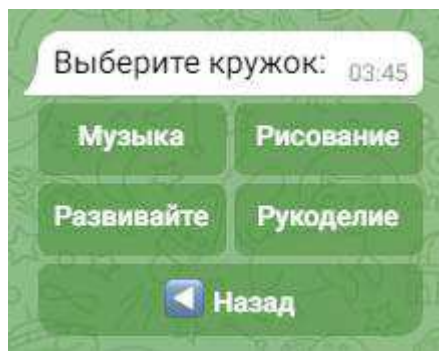


Рисунок 17 – Меню кружков

Программный код создания данного меню представлен на рисунке 18.

```
@dp.callback_query_handler(Text('lessons'))
async def select_lesson(call: types.CallbackQuery):
    '''Выбор занятия'''
    await call.answer()

    keyboard = types.InlineKeyboardMarkup(row_width=2)
    keyboard.add(
        types.InlineKeyboardButton('Музыка', callback_data='lesson_music'),
        types.InlineKeyboardButton('Рисование', callback_data='lesson_draw'),
        types.InlineKeyboardButton('Развивайте',
            callback_data='lesson_develop'),
        types.InlineKeyboardButton('Рукоделие',
            callback_data='lesson_handmade'),
        types.InlineKeyboardButton('◀ Назад', callback_data='start')
    )

    await call.message.edit_text('Выберите кружок:', reply_markup=keyboard)
```

Рисунок 18 – Код меню кружков



Выбрав кружок, пользователю предоставляется информация по этому кружку:

- кто ведет;
- дни проведения;
- время;
- на какое время и день записана группа.

На рисунке 19 представлена информация по выбранному кружку.

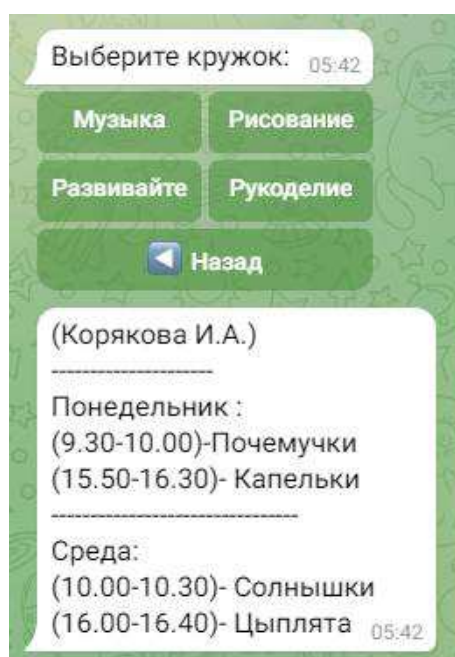


Рисунок 19 – Информация по кружку «Музыка»

На рисунке 20 представлен код вывода информации по кружку.

```
@dp.callback_query_handler(Text(startswith='lesson_'))
async def show_schedule(call: types.CallbackQuery, state: FSMContext):
    '''Показывает расписание кружка'''
    await call.answer()
    user_id = call.from_user.id
    lesson = call.data.split('_')[-1]
    await state.update_data(lesson=lesson)
    lesson_data = await getters.get_lesson_data(lesson)
```

Рисунок 20 – Вывод информации по кружку

В интерфейсе администратора добавляется кнопка «Добавить информацию». Если информация в этом разделе присутствует, то эта кнопка будет называться «Изменить информацию». Администратор может вставлять информацию в текущий раздел, нажимая на эту кнопку. На рисунке 21 демонстрируется добавление информации по кружку.

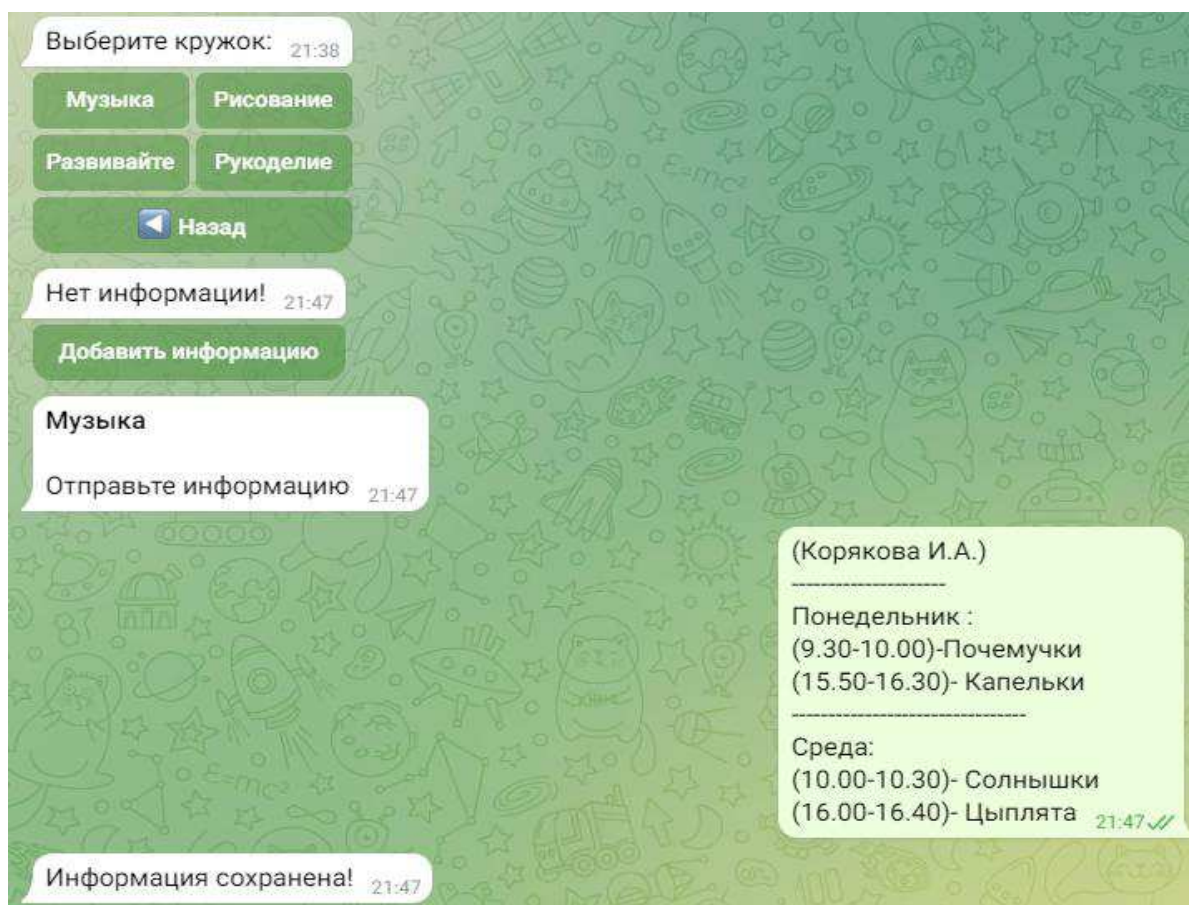


Рисунок 21 – Добавление информации по кружку

На рисунке 22 представлена часть кода для добавления информации со стороны администратора.

```

class AddLessonStates(StatesGroup):
    waiting_for_info = State()

    @dp.callback_query_handler(Text(startswith='info_add_'))
    async def add_info_lesson(call: types.CallbackQuery, state: FSMContext):
        '''Добавляет информацию о кружке'''
        await call.answer()
        lesson = call.data.split('_')[-1]
        await state.update_data(lesson=lesson)

        msg_text = (
            f'<b>{dictionary.lessons[lesson]}</b>\n\n'
            f'Отправьте информацию'
        )
        await call.message.answer(msg_text)
        await AddLessonStates.waiting_for_info.set()

    @dp.message_handler(state=AddLessonStates.waiting_for_info)
    async def save_info_lesson(message: types.Message, state: FSMContext):
        '''Сохранение информации'''
        state_data = await state.get_data()
        lesson = state_data['lesson']
        lesson_info = message.text
        await setters.add_lesson(lesson_info, lesson)

        await message.answer('Информация сохранена!')
        await state.reset_state(with_data=False)

```

Рисунок 22 – Часть кода добавления и сохранения информации по кружку

Для того, чтобы пользователь смог зайти в чат-бот как администратор, необходимо внести его IP Telegram в файл, где храниться токен чат-бота. На рисунке 23 представлен IP Telegram, сохраненный в файле.

```

TOKEN=5539171227:AAGnfG9pFESe80y6o92Fj2iUeSKt2t0iz-U
ADMINS=5198598721

```

Рисунок 23 - IP Telegram администратора

На рисунке 24 представлен код подключения администратора.

```
from environs import Env

env = Env()
env.read_env()

TOKEN = env.str('TOKEN')
ADMINS = [int(i) for i in env.list('ADMINS')]
```

Рисунок 24 - Код подключения администратора

Просмотрев информацию по кружкам, пользователь может вернуться в главное меню и выбрать «Группы», для дальнейшего просмотра расписания. Выбрав «Группы», пользователю предоставляется список групп в виде кнопок. Детский сад «Рябинка» состоит из четырех групп:

- цыплята;
- капельки;
- почемучки;
- солнышки.

На рисунке 25 представлено меню групп, составляющих детский сад «Рябинка».

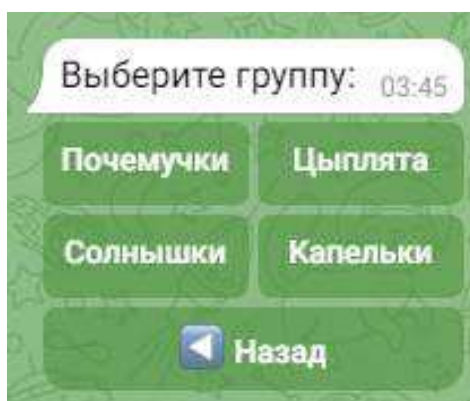


Рисунок 25 - Выбор группы

На рисунке 26 представлен код кнопок меню групп.

```

@dp.callback_query_handler(Text('groups'))
async def select_group(call: types.CallbackQuery):
    '''Выбор группы'''
    await call.answer()
    keyboard = types.InlineKeyboardMarkup(row_width=2)
    keyboard.add(
        types.InlineKeyboardButton('Почемучки',
            callback_data='group_pochemuchki'),
        types.InlineKeyboardButton('Цыплята', callback_data='group_cypljata'),
        types.InlineKeyboardButton('Солнышки', callback_data='group_solnyshki'),
        types.InlineKeyboardButton('Капельки', callback_data='group_kapelki'),
        types.InlineKeyboardButton('◀ Назад', callback_data='start'),
    )

    await call.message.edit_text('Выберите группу:', reply_markup=keyboard)

```

Рисунок 26 – Код кнопок меню групп

Далее идет выбор расписания. Расписание можно выбрать и посмотреть на:

- Сегодня – выводится расписание по календарному дню;
- Завтра – выводится расписание на следующий день по календарю.
- День недели – выводится расписание на выбранный день
- Все расписание – выводится расписание на все дни недели.

На рисунке 27 представлено меню выбора расписания.

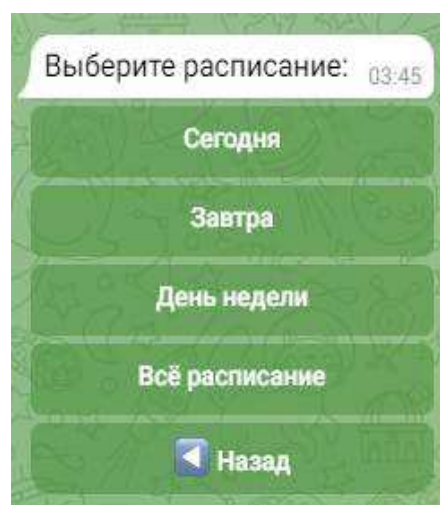


Рисунок 27 - Выбор расписания

На рисунке 28 представлен программный код выбора расписания.

```

@dp.callback_query_handler(Text(startswith='group_'))
async def select_schedule(call: types.CallbackQuery, state: FSMContext):
    '''Выбор расписания'''
    await call.answer()
    group = call.data.split('_')[-1]
    await state.update_data(group=group)

    keyboard = types.InlineKeyboardMarkup(row_width=1)
    keyboard.add(
        types.InlineKeyboardButton('Сегодня', callback_data='schedule_today'),
        types.InlineKeyboardButton('Завтра', callback_data='schedule_tomorrow'),
        types.InlineKeyboardButton('День недели', callback_data='schedule_day'),
        types.InlineKeyboardButton('Всё расписание',
            callback_data='schedule_all'),
        types.InlineKeyboardButton('◀ Назад', callback_data='groups'),
    )

    await call.message.edit_text('Выберите расписание:', reply_markup=keyboard)

```

Рисунок 28 – Код выбора расписания

Выбрав день, расписание, которое необходимо посмотреть, бот отправляет данное расписание. На рисунке 29 представлена часть выбранного расписания.

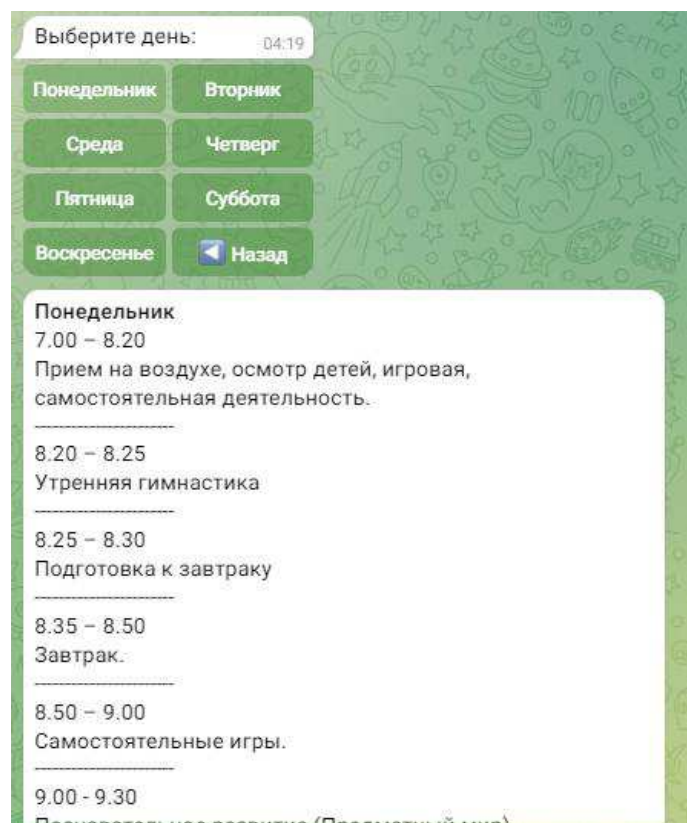


Рисунок 29 – Вывод расписания

На рисунке 30 представлена часть кода вывода расписания.

```
@dp.callback_query_handler(Text(startswith='day_'))
async def show_schedule_day(call: types.CallbackQuery, state: FSMContext):
    '''Показывает расписание на определённый день'''
    await call.answer()
    user_id = call.from_user.id
    day = call.data.split('_')[-1]
    state_data = await state.get_data()
    group = state_data['group']
    schedule_data = await getters.get_schedule_data(group)
    schedule_day = schedule_data[day]

    if not schedule_day:
        if user_id in config.ADMINS:
            keyboard = types.InlineKeyboardMarkup()
            keyboard.add(
                types.InlineKeyboardButton('Добавить расписание',
                    callback_data=f'add_{group}_{day}')
            )
        else:
            keyboard = None

        await call.message.answer('Нет расписания!', reply_markup=keyboard)
    else:
        if user_id in config.ADMINS:
            keyboard = types.InlineKeyboardMarkup()
            keyboard.add(
                types.InlineKeyboardButton('Изменить расписание',
```

Рисунок 30 – Код вывода расписания, лист 1

```
                callback_data=f'edit_{group}_{day}')
            )
        else:
            keyboard = None

        await call.message.answer(schedule_data[day],
            reply_markup=keyboard)
```

Рисунок 30 лист 2

Администратор со своей стороны, может, как вводить данное расписание (если расписание отсутствует), так и изменять его (рисунок 31).

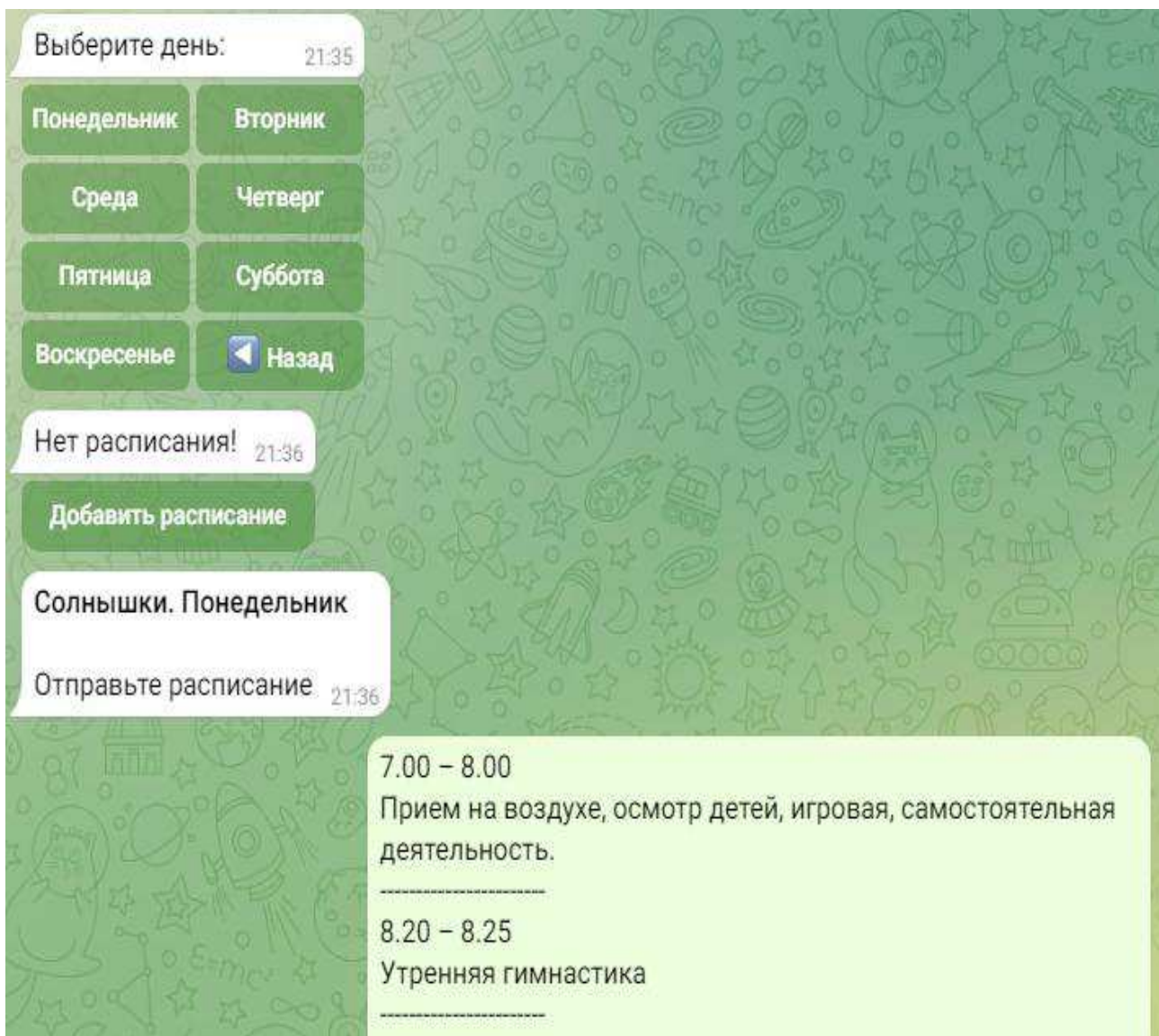


Рисунок 31 – Добавление расписания

На рисунке 32 представлен код для добавления и изменения расписания.



```

@dp.callback_query_handler(Text(startswith='add_'))
async def add_schedule(call: types.CallbackQuery, state: FSMContext):
    '''Добавляет расписание'''
    await call.answer()
    group, day = call.data.split('_')[1:]
    await state.update_data(group=group, day=day)

    msg_text = (
        f'<b>{dictionary.groups[group]}. {dictionary.days[day]}</b>\n\n'
        f'Отправьте расписание'
    )
    await call.message.answer(msg_text)
    await AddScheduleStates.waiting_for_schedule.set()

@dp.message_handler(state=AddScheduleStates.waiting_for_schedule)
async def save_schedule(message: types.Message, state: FSMContext):
    '''Сохранение расписания'''
    state_data = await state.get_data()
    day = state_data['day']
    group = state_data['group']
    schedule = message.text
    await setters.add_schedule(day, group, schedule)

    await message.answer('Расписание сохранено!')
    await state.reset_state(with_data=False)

```

Рисунок 32 – Код добавления и изменения расписания

Для хранения и изменения информации используется база данных, созданная в SQLite. Подключившись к данной базе данных, создаем необходимую БД и таблицы в ней при запуске бота. На рисунке 33 представлен код создания БД и таблиц в ней при запуске бота.

```

from aiogram import executor

import handlers
from loader import dp
from database import getters
from database.work_db import create_tables, placeholder_tables

async def on_startup(dp):
    '''Действия при запуске бота'''
    print('Бот запущен!')

    # Создание БД и таблиц в ней
    await create_tables()

    schedule_all = await getters.get_schedule_all()
    lessons = await getters.get_lessons()

    if not schedule_all:
        await placeholder_tables(groups_table=True)

    if not lessons:
        await placeholder_tables(lessons_table=True)

if __name__ == '__main__':
    executor.start_polling(dp, on_startup=on_startup)

```

Рисунок 33 – Код создания БД и таблиц при запуске бота

Создание необходимых полей таблиц в базе данных представлено на рисунке 34.

```
async def create_tables():
    '''Создание таблиц в БД'''
    groups = '''
CREATE TABLE IF NOT EXISTS groups (
    name TEXT,
    monday TEXT,
    tuesday TEXT,
    wednesday TEXT,
    thursday TEXT,
    friday TEXT,
    saturday TEXT,
    sunday TEXT
);
'''

    lessons = '''
CREATE TABLE IF NOT EXISTS lessons (
    lesson_name TEXT,
    info TEXT
);
'''

    await execute_query(groups)
    await execute_query(lessons)
```

Рисунок 34 – Код создания таблиц в БД

Программный код для добавления информации в базу данных представлен на рисунке 35.

```

from .work_db import *
from database import keys

async def add_schedule(day, group, schedule):
    '''Добавляет расписание в БД'''
    query = '''
UPDATE
    groups
SET
    {} = ?
WHERE
    name = ?
    '''.format(day)

    await execute_query(query, [schedule, group])

async def add_lesson(lesson_info, lesson):
    '''Добавляет информацию о кружке в БД'''
    query = '''
UPDATE
    lessons
SET
    info = ?
WHERE
    lesson_name = ?
    '''

    await execute_query(query, [lesson_info, lesson])

```

Рисунок 34 – Код добавления информации в БД

Часть программного кода, для возврата расписания групп и кружков, представлена на рисунке 36. Аналогичный код идет для возвращения другой информации.

```

async def get_schedule_data(group):
    '''Возвращает расписание'''
    query = '''
SELECT
    *
FROM
    groups
WHERE
    name = ?
    '''

    schedule_data = await execute_query(query, [group], fetch='one')

    if schedule_data:
        return dict(zip(keys.groups, schedule_data))

async def get_schedule_all(group=None):
    '''Возвращает всё расписание для всех групп'''
    if group:
        query = '''
SELECT
    *
FROM
    groups
WHERE
    name = ?
    '''

```

Рисунок 36 – Часть программного кода для возврата расписания, лист 1

```

        schedule_all = await execute_query(query, [group], fetch='all')
    else:
        query = '''
SELECT
    *
FROM
    groups
    '''

        schedule_all = await execute_query(query, fetch='all')

    if schedule_all:
        return [dict(zip(keys.groups, i)) for i in schedule_all]

```

Рисунок 36, лист 2

## **2.4 Выводы по разделу «Разработка чат бота «Расписание МБДОУ «Рябинка»»**

Разработан чат-бот «Расписание МБДОУ «Рябинка». Данный бот содержит расписание каждой группы на каждый день недели, а также информацию по проводимым кружкам в муниципальном бюджетном дошкольном образовательном учреждении детском саду «Рябинка».

Разработанный чат-бот позволяет использовать две роли:

- пользователь;
- администратор.

Пользователь может просмотреть информацию о кружках, имеющихся в детском саду, или выбрать и посмотреть расписание у любой группы. Администратор же в свою очередь может добавлять или изменять любую информацию, как в кружках, так и в расписании групп.

## **3 Расчет затрат и оценка экономической эффективности чат-бота «Расписание МБДОУ «Рябинка»»**

### **3.1 Расчет затрат на разработку чат-бота «Расписание МБДОУ «Рябинка»»**

Для расчета совокупности затрат на внедрение проекта необходимо использовать методику, которая могла бы собрать все данные вместе и предоставлять возможность для изучения рентабельности внедрения. Такой является популярная методика ТСО (Total cost ownership) – совокупная стоимость владения. Данный метод предполагает количественную оценку на внедрения и сопровождение программного обеспечения.

Затраты реализации можно рассчитать по методике ТСО, которая рассчитывается по формуле

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

где DE – прямые затраты;

IC – косвенные затраты первой и второй группы.

### 3.2 Капитальные затраты

Капитальные затраты вычисляются по следующей формуле

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оэ}}, \quad (2)$$

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

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

$K_{\text{лс}}$  – 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;

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

Затраты на проектирование рассчитываются по формуле

$$K_{\text{пр}} = K_{\text{зп}} + K_{\text{инс}} + K_{\text{свт}} + K_{\text{проч}}, \quad (3)$$

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

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

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

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

Рассчитаем зарплату программиста. Оклад работы программиста за месяц равен 15 000 рублей.

Оклад уплачивается в зависимости от количества рабочих дней в расчетном месяце, например, в марте 22 рабочих дня, т.к. программист работает 21 день, следовательно, заработная плата программиста равна 14 318 руб. (без учета северного коэффициента, региональной надбавки и НДФЛ).

Зарботная плата программиста, с учетом северного коэффициента и региональной надбавки, рассчитана по тарифному способу оплаты. Расчет зарплаты программиста представлен в таблице 6.

Таблица 6– Расчет зарплаты программиста

Наименование составляющих зарплаты	Числовые показатели, руб.
Оклад	14 318
Северный коэффициент	4 295,4.
Районный коэффициент	4 295,4.
НДФЛ	2 457
Выплаты во внебюджетные фонды	6 918
Итого	29 826,8.

$$K_{\text{зп}} = 29\,826,8 \text{ руб.}$$

Расчет стоимости программного обеспечения для создания информационной системы представлен в таблице 7.

Таблица 7 – Затраты на ПО

Наименование программного обеспечения	Количество единиц	Срок использования	Цена, руб.
Microsoft Windows 10 Professional	1	Бессрочно	14 800
Облачный сервер	1	Год	1 899
Итого			16 699

Microsoft Windows 10 Professional стоит 14 800 руб., использоваться будет в течении пяти лет. Использоваться в данной системе он будет 21 день.



$$A_{пр} = \frac{A_{год}}{K_{дг}} * K_{дэ} , \quad (4)$$

где  $A_{год}$  – амортизация за год использования;

$A_{пр}$  – проектная амортизация;

$K_{дг}$  – количество дней в году;

$K_{дэ}$  – количество дней эксплуатации.

$$A_{год} = 14\,800 / 5 = 2\,960 \text{ руб.}$$

$$A_{пр} = \frac{2\,960}{365} * 21 = 170 \text{ руб.}$$

Стоимость web-хостинга и домена учитываются в проекте полностью, также в эту сумму включается место для хранения данных на диске.

$$K_{ипс} = 2\,069 \text{ руб.}$$

Для реализации проекта необходим компьютер для программиста, составляющие компьютера представлены в таблице 8.

Таблица 8 – Компьютер для программиста

Наименование оборудования	Наименование элементов оборудования	Количество единиц	Стоимость единицы, руб.	Стоимость всего, руб.
Компьютер для разработчика	Процессор Intel Core i5-9400	1	14000	14000
	Видеокарта Nvidia GeForce GTX 1050 (2Gb)	1	7000	7000
	1 ТБ Жесткий диск WD Purple [WD10PURZ]	1	3700	3700

Продолжение таблицы 8

	Оперативная память AMD Radeon R5 Entertainment Series [R538G1601U2S-U]	1	4 500	4 500
	Блок питания DEXP DPS-400	1	1 450.	1 450
	Корпус DEXP AWS- DE7	1	1 550	1 550
	Монитор DEXP FF220K	1	6700	6700
	Клавиатура DEXP KB0203	1	600	600
Итоговая стоимость				39 850

**Расчет амортизации.** По статистике срок службы оборудования составляет:

- Оперативная память - 10 лет.
- Материнская плата - 5 лет.
- Жесткий диск HDD - 5 лет
- Жесткий диск SSD - 8 лет.
- Процессор - 10 лет.
- Видеокарта - 4 года.
- Блок питания - 10 лет.
- Корпус - 3 года.
- Монитор - 5 лет.
- Клавиатура - 3 года.
- Мышь - 3 года.

При нормальной эксплуатации компьютер прослужит 6 лет.

Амортизация рассчитывается по формуле

$$A_{\text{год}} = C_{\text{б}} * H_{\text{ам}}, \quad (5)$$

где  $A_{\text{год}}$  – амортизация за год использования;

$C_{\text{б}}$  – балансовая стоимость;

$H_{\text{ам}}$  – норма амортизации.

Проектная амортизация рассчитывается по формуле

$$A_{\text{пр}} = \frac{A_{\text{год}}}{K_{\text{рдг}}}, \quad (6)$$

где  $A_{\text{год}}$  – амортизация за год использования;

$A_{\text{пр}}$  – проектная амортизация;

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

$K_{\text{дэ}}$  – количество дней эксплуатации.

Средний срок эксплуатации компьютера равен 6-ти годам, следовательно, норма амортизации равна 16,67% по формуле

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

где  $H_{\text{ам}}$  – норма амортизации;

$T_{\text{эк}}$  – срок эксплуатации компьютера.

$$H_{\text{ам}} = \frac{100\%}{6} = 16,67\%.$$

$$A_{\text{год}} = 39\,850 * 16,67 = 6\,643 \text{ руб.}$$

$$A_{\text{пр}} = \frac{6\,643}{247} * 21 = 564,8 \text{ руб.}$$

$$K_{\text{свт}} = 564,8 \text{ руб.}$$

Возможен риск, что в ходе работы какой-либо из компонентов выйдет из строя, необходимо иметь резерв в виде 1500 руб. на возможный ремонт или замену комплектующих.

$$K_{\text{проч}} = 1500 \text{ руб.}$$

$K_{\text{зп}} = 29\,826,8$  руб.– затраты на заработную плату проектировщиков;

$K_{\text{ипс}} = 2\,069$  руб.– затраты на инструментальные программные средства для проектирования;

$K_{\text{свт}} = 564,8$  руб.– затраты на средства вычислительной техники для проектирования;

$K_{\text{проч}} = 1500$  руб.– прочие затраты на проектирование.

$$K_{\text{пр}} = 29\,826,8 + 2\,069 + 564,8 + 1500 = 33\,960,6 \text{ руб.}$$

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оз}}, \quad (8)$$

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

$K_{\text{тс}}$  – затраты на технические средства управления, равны 0, так как используем ПК, который имеется в учреждении;

$K_{\text{лс}}$  – затраты на создание линий связи локальных сетей, равны 0, так как при проектировании и разработке чат-бота используется локальный компьютер;

$K_{\text{по}}$  – затраты на программные средства, равны 0, так как особые средства для работы системы не нужны, будет установлен Telegram, он бесплатный, остальные предустановлены на компьютере который есть в учреждении;

$K_{\text{ин}}$  – затраты на формирование информационной базы. Воспитатель будет вносить изменение в расписание, затраты равны 0, так как он будет делать это бесплатно;

$K_{\text{об}}$  – затраты на обучение персонала. Необходимо объяснить воспитателям, как работать с продуктом, это займет часа 3.  $K_{\text{об}}$  это зарплата разработчика за 3 часа.

$$K_{\text{об}} = 14\,318 / 21 / 8 * 3 = 255 \text{ руб.}$$

$K_{\text{оп}}$  – затраты на опытную эксплуатацию. Тестирование продукта суммарно займет всего 10 дней, по 2 часа, это 20 часов или 3 рабочих дня.

$$K_{\text{оп}} = 2\,250 \text{ руб.}$$

$$K = 36\,465,6 \text{ руб.}$$

Список капитальных затрат представлен в таблице 9.

Таблица 9 – Список капитальных затрат

Затраты	Состав затрат	Планируемая сумма, руб.
Затраты на разработку ПП	Затраты на заработную плату разработчика	29 826,8
	Затраты на программные средства	2 069
	Затраты на средства вычислительной техники	564,8
	Прочие затраты на разработку	1500
	Затраты на обучение персонала	255
	Затраты на опытную эксплуатацию	2 250
Итого		36 465,6

Диаграмма капитальных затрат изображена на рисунке 37.



Рисунок 37 – Диаграмма «Капитальные затраты»

### 3.3 Расчет эксплуатационных затрат

Расчет эксплуатационных затрат производится по формуле

$$C = C_{зп} + C_{ао} + C_{то} + C_{лс} + C_{ни} + C_{проч}, \quad (9)$$

где  $C_{зп}$  – зарплата персонала, работающего с программным продуктом:

$C_{ао}$  – амортизационные отчисления;

$C_{то}$  – затрата на техническое обслуживание;

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

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

$C_{проч}$  – прочие затраты.

**Затраты на заработную плату персонала.** Обслуживанием и внесением данных по расписанию будет заниматься воспитатель. Работа будет занимать всего один час в месяц, по итогу 12 часов в год. Обслуживание ИС, будет стоит зарплате воспитателя за час.

$$28\,305 / 22 / 12 = 107 \text{ руб./час}$$

$$C_{\text{зп}} = 107 * 12 = 1\,284 \text{ руб.}$$

**Затраты на амортизационные отчисления.** Для применения ПП будет использоваться устройство пользователя, следовательно, расчет амортизационных затрат на эксплуатацию не требуется.

$$C_{\text{ао}} = 0 \text{ руб.}$$

**Затраты на техническое обслуживание.** Затраты на обслуживание и ремонт не учитываются, так как ремонт устройств пользователей не входит в услуги, предоставляемые ПП, следовательно, эксплуатационные затраты на техническое обслуживание равен 0.

$$C_{\text{то}} = 0 \text{ руб.}$$

**Затраты на использование глобальных сетей.** Пользование глобальными сетями в рамках эксплуатации проекта не учитывается, т.к. пользователи ПП будут пользоваться имеющимся у них интернетом, из этого следует, что эксплуатационные затраты на использование глобальных сетей равен 0.

$$C_{\text{лс}} = 0 \text{ руб.}$$

**Затраты на носители информации.** На первый год, цена облачного сервера учтена в пункте 3.2 на этапе внедрения, последующий год будет такая цена, сейчас 0.

$$C_{\text{ни}} = 0 \text{ руб.}$$

**Прочие затраты.** Прочие затраты равны 5% от суммы всех затрат.

$$C_{\text{проч}} = 1\,284 * 0,05 = 64,2 \text{ руб.}$$

$$C = C_{\text{зп}} + C_{\text{ао}} + C_{\text{то}} + C_{\text{лс}} + C_{\text{ни}} + C_{\text{проч}}, \quad (10)$$

где  $C_{\text{зп}} = 1\,284$  руб. – зарплата персонала, работающего с программным продуктом;

$C_{\text{ао}} = 0$  – амортизационные отчисления;

$C_{\text{то}} = 0$  – затрата на техническое обслуживание;

$C_{\text{лс}} = 0$  – затраты на использование глобальных сетей;

$C_{\text{ни}} = 0$  – затраты на носители информации;

$C_{\text{проч}} = 64,2$  руб. – прочие затраты.

$$C = 1\,284 + 64,2 = 1\,348,2 \text{ руб.}$$

Эксплуатационные затраты представлены в таблице 10.

Таблица 10 – Эксплуатационные затраты

Состав затрат	Планируемая сумма, рублей в год
Затраты на заработную плату персонала	1 284
Затраты на амортизацию отчисления	0
Затраты на техническое обслуживание	0
Затраты на использование глобальных сетей	0
Затраты на носители информации	0
Прочие затраты	64,2
Итого	1 348,2

Состав эксплуатационных затрат изображен на рисунке 38.



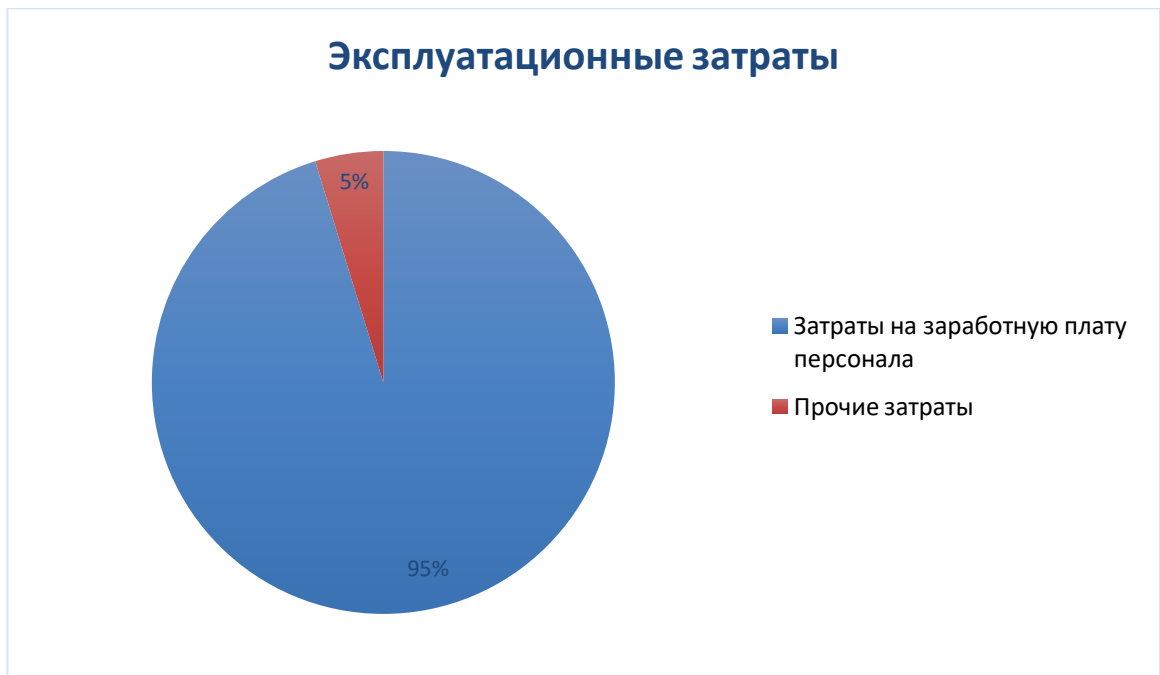


Рисунок 38 – Эксплуатационные затраты

### 3.4 Расчет совокупной стоимости владения системой

Прямые затраты рассчитываются по формуле

$$DE = DE_1 + DE_2 + DE_3 + DE_4 + DE_5 + DE_6 + DE_7 + DE_8, \quad (11)$$

где  $DE_1$  – капитальные затраты;

$DE_2 = 1\,348,2$  – расходы на управление информационными технологиями;

$DE_3 = 0$  – расходы на техническую поддержку автоматизированного обеспечения и программного обеспечения;

$DE_4 = 0$  – расходы на разработку прикладного программного обеспечения внутренними силами, нет необходимости разрабатывать ПО;

$DE_5 = 0$  – расходы на аутсорсинг, не требуются внешние источники;

$DE_6 = 0$  – командировочные расходы;

$DE_7 = 0$  – расходы на услуги связи, все необходимое для выхода в интернет уже имеется;

$DE_8 = 1500$  – другие группы расходов.

Прямые затраты:

$$DE = 36\,465,6 + 1\,348,2 + 1\,500 = 39\,313,7 \text{ руб.}$$

Совокупные затраты владения продуктом будут равны:

$$TCO = 39\,313,7 + 0 + 1\,284 = 40\,597,7 \text{ руб.}$$

### **3.5 Экономическая эффективность реализации проекта**

Чат-бот «Расписание МБДОУ «Рябинка» не является программным продуктом для получения прибыли. Он поможет людям сократить время на поиски необходимого расписания, предоставит информацию по кружкам, имеющимся в МБДОУ «Рябинка», а также даст возможность родителям посмотреть расписание своей группы и других групп.

Эффективность программного продукта будет в быстром доступе необходимой информации для сотрудников МБДОУ «Рябинка» и родителей. Тем самым сократив время на поиски расписания в бумажном варианте.

Являясь социальным проектом, он не несет за собой никакой экономической эффективности, ведь он не сможет покрыть затраты на реализацию проекта.

### **3.6 Оценка риска при реализации проекта создания программного продукта**

**Идентификация рисков.** Анализ рисков является одним из решающих факторов успешности реализации проекта на рынке. Рассмотрев данный проект и ему подобные, были идентифицированы следующие риски:

- Риск возможности закрытия площадки или сервера.
- Изменение требований заказчика к проекту.

- Лимитированное время разработки.
- Риск соответствия.
- Риск денежных потоков.

**Анализ рисков.** Более подробно рассмотрим каждый из рисков.

Риск возможности закрытия площадки или сервера – данный риск возникает в случаях, если площадка или сервер на котором создавался продукт, закроются или перестанут работать.

Изменение требований заказчика к проекту – внесение заказчиком новых пожеланий и требований. Исходя из этого риска, могут «вытекать» другие риски: лимитированное время разработки и риск соответствия.

Лимитированное время разработки – подобный риск может возникнуть, если понадобится перенос программного продукта на другую платформу (WhatsApp, Viber и др.).

Риск соответствия – заключается в том, чтобы программный продукт соответствовал техническому заданию.

Риск денежных потоков – данный риск возникает в случаях несвоевременной оплаты проекта заказчиком.

Таблица 11 – Анализ рисков

№	Группы рисков	Перечень рисков проекта	Уровень влияния риска на проект	Вероятность риска	Возможность снижения риска
1	Риск возможности закрытия площадки или сервера	Закрытие площадки или сервера	Средний	Средний	Подбирать площадку или сервер так, чтобы в случае чего, перейти на другую
2	Изменение требований заказчика к проекту	Внесение заказчиком изменений в требования к проекту	Средний	Средний	Заранее обговорить и документально заверить установленные требования

Продолжение таблицы 11

3	Риск соответствия	Несоответствие техническому заданию	Средний	Низкая	Установление четкого соответствия требований к стратегическим целям
4	Риск денежных потоков	Несвоевременная оплата проекта	Средний	Низкая	Договориться с заказчиком о предоплате, либо об оплате проекта по мере реализации

### 3.7 Выводы по разделу «Расчет затрат и оценка экономической эффективности чат-бота «Расписание МБДОУ «Рябинка»»

В данном разделе были представлены расходы для создания данного программного продукта. Рассчитаны капитальные и эксплуатационные затраты, которые входят в группу прямых затрат. Также произведен расчет ТСО.

Капитальные затраты составили  $K = 36\,465,6$  рублей

Эксплуатационные затраты составили  $C = 1\,348,2$  рублей.

Прямые затраты составили  $DE = 39\,313,7$  рублей.

Стоимость годовой амортизации  $ТСО = 40\,597,7$  рублей.

Определена эффективность данного проекта. Рассмотрены возможные риски для проекта и найдены способы их решения.

## ЗАКЛЮЧЕНИЕ

В процессе выполнения выпускной квалифицированной работы был проведен анализ текущей деятельности МБДОУ «Рябинка», выявлена актуальность разработки чат-бота, а также выбрана платформа Telegram для размещения чат-бота.

Во втором разделе было произведено проектирование и разработка чат-бота «Расписание МБДОУ «Рябинка».

Третий раздел содержит в себе информацию о расчете затрат и оценке экономической эффективности данного программного продукта.

Результатом работы является спроектированный и разработанный чат-бот «Расписание МБДОУ «Рябинка», позволяющий сократить временные затраты на поиски необходимого расписания и получении информации по имеющимся кружкам в МБДОУ «Рябинка».

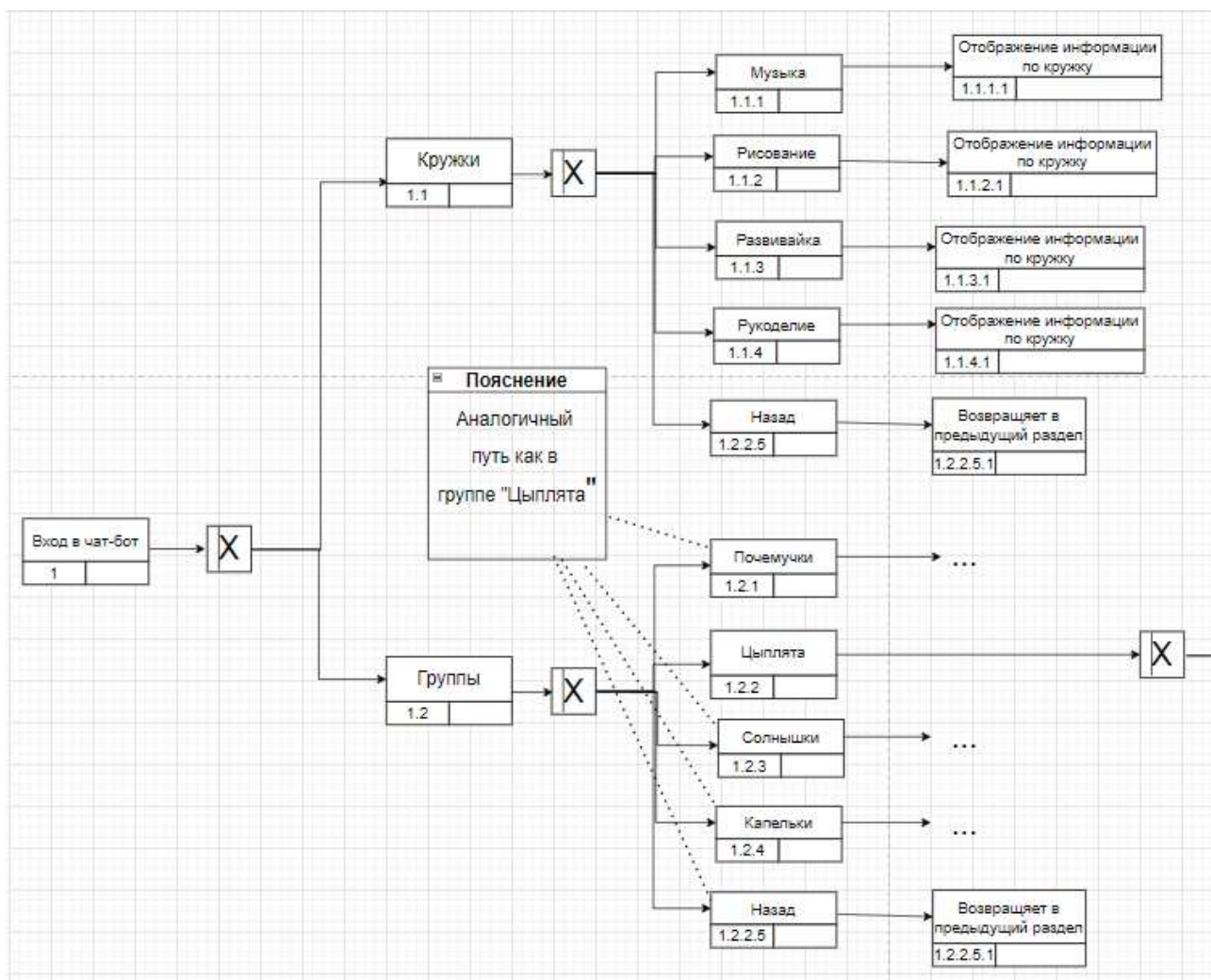
## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Деятельность детского сада / ЗАТО Озерный : [сайт]. – URL: <https://detsad1-zatolake.edusite.ru/p4aa1.html> (дата обращения: 22.03.22).
2. Выбираем мессенджер / CMS magazine все оDIGITAL : [сайт]. – URL: <https://cmsmagazine.ru/journal/items-we-choose-the-messenger/> (дата обращения: 09.03.22).
3. Выбираем самый удобный редактор кода Python / Харб : [сайт]. – URL: <https://habr.com/ru/company/skillfactory/blog/521838/> (дата обращения: 13.03.22).
4. Образование / МБДОУ «Рябинка» : [сайт]. – URL: <https://clck.ru/eiWq8> (дата обращения: 20.05.22).
5. Основные сведения / МБДОУ «Рябинка» : [сайт]. – URL: <https://clck.ru/eiWq8> (дата обращения: 22.05.22).
6. Публичный отчет / Pandia : [сайт]. – URL: <https://pandia.ru/text/78/090/46120.php> (дата обращения: 28.05.22).
7. Плюсы и минусы языков программирования / TELETYPE : [сайт]. – URL: [https://teletype.in/@itvdn\\_it/zDfuiuHU](https://teletype.in/@itvdn_it/zDfuiuHU) (дата обращения: 30.03.22).
8. Проектирование UseCase диаграммы / worldskills Russia : [сайт]. – URL: <https://clck.ru/qs6Sq> (дата обращения: 30.04.22).
9. Структура ДОУ / Чиполино : [сайт]. – URL: <http://www.chipol34.caduk.ru/p28aa1.html> (дата обращения: 18.03.22).
10. Структура и органы управления образовательной организацией / МБДОУ «Рябинка» : [сайт]. – URL: <https://clck.ru/eiWq8> (дата обращения: 20.03.22).
11. СТУ 7.5-07-2021 СТАНДАРТ УНИВЕРСИТЕТА: Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. – URL: <https://clck.ru/ahY34> (дата обращения: 26.03.22).
12. Хостинг или VPS/VDS / Макхост : [сайт]. – URL:

- <https://mchost.ru/articles/что-лучше-hosting-или-vps/> (дата обращения: 25.03.22).
13. Что лучше хостинг или VPS / Shneider : [сайт]. – URL: <https://clck.ru/qsZPA> (дата обращения: 25.03.22).
14. DFD / Википедия : [сайт]. – URL: <https://ru.wikipedia.org/wiki/DFD> (дата обращения: 26.03.22).
15. Masterhost (Мастерхост) / Хостинг в деталях отзывы и обзоры в хостингах : [сайт]. – URL: <https://hosting101.ru/masterhost.ru-vps> (дата обращения: 13.03.22).
16. Mchost.ru / Хостинг в деталях отзывы и обзоры в хостингах : [сайт]. – URL: <https://hosting101.ru/mchost.ru-vps> (дата обращения: 13.03.22).
17. MongoDB / SkillFactory.Блог : [сайт]. – URL: <https://blog.skillfactory.ru/glossary/mongodb/> (дата обращения: 24.03.22).
18. MySQL / WEB CREATOR : [сайт]. – URL: <https://web-creator.ru/articles/mysql> (дата обращения: 03.06.22).
19. Reg.ru / Хостинг в деталях отзывы и обзоры в хостингах : [сайт]. – URL: <https://hosting101.ru/reg.ru-vps> (дата обращения: 13.03.22).
20. Reg.ru / Википедия : [сайт]. – URL: <https://ru.wikipedia.org/wiki/Reg.ru> (дата обращения: 14.04.22).
21. SQLite / Лаборатория линуксоида : [сайт]. – URL: <https://younglinux.info/sqlite/sqlite> (дата обращения: 03.03.22).

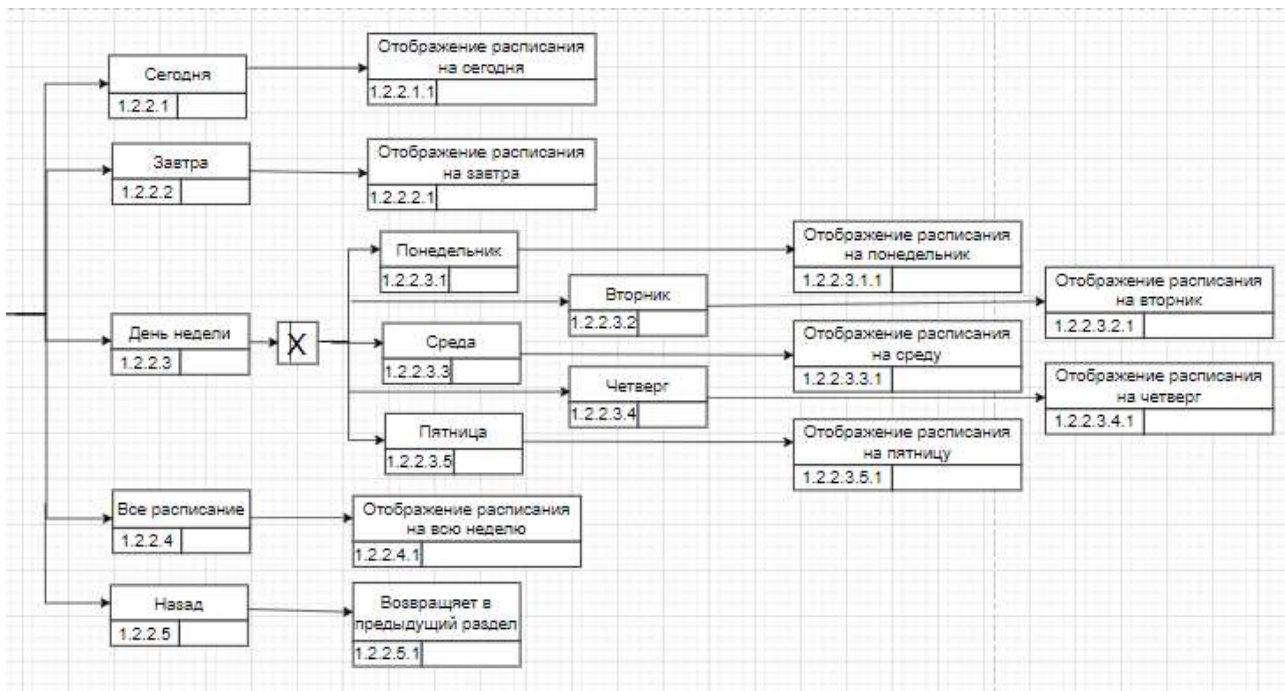
# ПРИЛОЖЕНИЕ А

## Модель работы пользователя с чат-ботом



Рисунгой А.1 – Последовательно меню чат-бота





Рисуной А.1, лист 2

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 21 наименование.

Один экземпляр сдан на кафедру.

« \_\_\_\_ » \_\_\_\_\_ 2022 г.

\_\_\_\_\_ Галицына Татьяна Евгеньевна  
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, математики и естественно-научных  
дисциплин

УТВЕРЖДАЮ

Заведующий кафедрой

  
Е. Н. Скуратенко

подпись


« 17 » 06 2022 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.03 Прикладная информатика


Разработка чат-бота «Расписание МБДОУ «Рябинка»

Руководитель

  
17.06.2022  
подпись, дата

доцент, канд. физ.-мат. наук А. Н. Ташкин

Выпускник

  
17.06.2022  
подпись, дата

Т. Е. Галицына

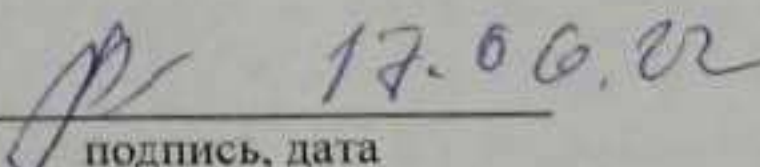
Консультанты  
по разделам:

Экономический

  
17.06.22  
подпись, дата

Е. Н. Скуратенко

Нормоконтролер

  
17.06.22  
подпись, дата

В. И. Кокова

Абакан 2022