Индекс УДК 004.584

Андреева М.М. Andreeva M.M.

казанский национальный исследовательский технический университет им. а. н. туполева, 420111 г. казань, ул. к. маркса, 10

kazan national research technical university named after a. n. tupolev, 420111, kazan, 10 k.marx st.

e-mail: [andreeva-mm03@yandex.ru](mailto:andreeva-mm03@yandex.ru)

Научный руководитель: Р.М. Шакирзянов, к.т.н. Федеральное государственное бюджетное образовательное учреждение высшего образования «Казанский национальный исследовательский технический университет им. А.Н.Туполева-КАИ»

**Telegram-бот для абитуриентов КНИТУ-КАИ**

**Telegram-bot for KNRTU-KAI enrollee**

Описана система из двух Telegram-ботов для коммуникации с абитуриентами. Один бот предоставляет информацию об университете, а также позволяет пользователю создать личный кабинет и следить за своими баллами. Второй бот предназначен для ответов на вопросы пользователей, имеет функцию создания суперчата для использования несколькими модераторами.

Two Telegram-bots have been developed for communication with enrollees. One bot provides information about the university, and allows users to create a personal account and monitor their scores. The second bot is developed to answer user questions and has the function of creating a super chat for use by several moderators.

Python, Telegram, bot, Питон, Телеграм, Бот

**Цель:** предоставить возможность пользователям Telegram с помощью бота получать информацию о поступлении в университет.

**Задачи:**

1. Проработать информацию с сайта КНИТУ-КАИ, чтобы уместить данные в формат текстового сообщения.
2. Разделить информацию с сайта по темам и ролям.
3. Создать структуру бота, позволив клиенту выбирать роль и тему с помощью клавиатуры.
4. Реализовать функцию «Личный кабинет», используя базы данных для хранения информации о пользователях.
5. Реализовать функцию просмотра проходных баллов по институтам.
6. Провести поиск наиболее удобного способа реализации crm платформы внутри Telegram.
7. Добавить возможность задать вопрос, создать удобный способ ответа для модераторов.
8. Проверить и, при необходимости, дооснастить имеющуюся IT-инфраструктуру для функционирования продукта.
9. Проверить ботов на пользователях и убедиться в интуитивности интерфейса.
10. Обеспечить беспрерывную работу бота на серверах.
11. Обеспечить сопровождение функционирования и поддержку пользователей в чате.

**Проблематика:** Распространение информации в последние десятилетия эффективно осуществляется посредством интернета. В настоящее время широкую получили популярность Telegram-боты, позволяющие не только довести до пользователя информацию о предложениях в структурированном виде, аналогично сайту, но и обеспечить интерактивное взаимодействие с пользователем.

Для удобства абитуриентов есть смысл внедрить различные сервисы для удобного информирования и взаимодействия с как можно большим количеством пользователей за счёт дублирования функциональности разными инструментами, такими как сайт, Telegram-бот, личное взаимодействие.

В настоящий момент взаимодействие организовано только посредством сайта и личного взаимодействия. В то же время мессенджер Telegram активно развивается, все больше людей регистрируется на этой удобной платформе. Мессенджер предоставляет возможность создания ботов, которые предоставляют дружелюбный и интуитивный интерфейс и позволяют направлять пользователю информацию в личные сообщения. Таким образом абитуриент может узнать информацию не прерывая общения на платформе, а также пропустит намного меньше информации, благодаря уведомлениям.

Необходимо разработать Telegram-бота, дублирующего информацию с сайта, предоставляющего возможности личного кабинета и консультирования пользователей посредством чата. Срок завершения разработки - май, чтобы всё лето бот продолжал стабильную работу.

**Критерии успеха:**

1. Разработан и внедрён Telegram-бот
2. Абитуриенты активно используют бота, заполняют личный кабинет и задают вопросы

**Заинтересованные стороны:**

1. Приёмная комиссия КНИТУ-КАИ – ожидают повышенный интерес абитуриентов к боту. Уменьшение количества вопросов, на которые ответ есть на сайте. Ожидают удобную платформу для ответов на вопросы абитуриентов внутри мессенджера Telegram.
2. Абитуриенты – ожидают удобного бота для просмотра информации, хранения своих баллов ЕГЭ, подсчёта суммы баллов и с функцией задавать вопросы напрямую приёмной комиссии.

**Эксперты проекта:**

1. Ответственный секретарь приемной комиссии, начальник управления организации приема абитуриентов

**Сроки:** Для реализации проекта отведено три месяца, поскольку к лету ожидается активная фаза использования. Таким образом проект должен быть запущен к концу мая. В процессе работы бота нужно контролировать жалобы абитуриентов, допустима модификация и изменение бота при получении новых запросов и жалоб. Бот к концу мая должен быть разработан и внедрён, сотрудники приёмной комиссии должны быть ознакомлены с суперчатом, создаваемым ботом, и понимать, как отвечать на вопросы пользователей. Абитуриентам должно быть сообщено о появлении Telegram-бота путём объявления в социальных сетях университета.

**Проект включает в себя:** Telegram-ботов, которые дублируют информацию с сайта, предоставляют функцию личного кабинета с баллами ЕГЭ и чат для ответов на вопросы.

**Бизнес-процесс устроен следующим образом:** пользователь переходит в разработанный Telegram-бот, выбирает, куда он хочет поступать, и получает доступ к информации об университете, направлениях, проходных баллах, важной информации, личному кабинету, в который он может ввести баллы для получения рекомендаций по выбору направления, а также ссылку на бота, который отвечает на вопросы.

**Риски проекта:**

1. Просрочка сроков. Необходимо быстро разработать ТЗ, оперативно вносить правки, чтобы обеспечить готовность проекта в срок.
2. Малая популярность бота у абитуриентов. Необходимо проинформировать о наличии бота в социальных сетях университета, а также предоставить пользователям интуитивный и удобный интерфейс, чтобы они продолжали пользовать ботом и рекомендовали его знакомым абитуриентам.

В ходе реализации проекта возможны следующие риски.

1. Временной риск – затягивание разработки по срокам больше, чем это было запланировано.
2. Внешние риски представлены такими, как, например, блокировка Telegram, и введение платы за работу бота.
3. Единая точка отказа – потеря интернет-соединения у сервера, отключение электричества, также форс-мажоры вроде пожара в офисе.
4. Сильных зависимостей у проекта с другими не наблюдается, т.к. он взаимодействует с имеющейся разработанной платформой, но могут возникнуть проблемы рассинхронизащии данных (требует аккуратного своевременного обновления информации) и неготовность сервера тянуть дополнительный сервис.
5. Риск изменения требований минимален, поскольку задача стоит чёткая, по оценке выполнимая. Появление необходимости внести дополнительную функциональность в бота – управляемая ситуация, это можно сделать в рамках нового проекта по его последующей доработке. На функциональность того, что запланировано, это не должно повлиять.

**Метрики проекта.**

1. Количество пользователей бота за всё время.
2. Количество пользователей бота за месяц и неделю.
3. Количество пользователей, заполнивших информацию о баллах ЕГЭ в личном кабинете.

**Требования к продукту.**

1. Актуальное информационное наполнение – информация должна полностью соответствовать той, что размещена на сайте и соответствует действительности.
2. Стабильность работы – бот должен работать, не допуская сбоев для обеспечения удовлетворённости использования клиентами.
3. Интуитивный интерфейс – пользователь должен легко понимать, как взаимодействовать с ботом.

**Для определения приоритетности задач представим их в виде модели MSCW**

M - must, обязательные:

1. Проработать логику работы бота, всё дерево его решений, выдачи информации.
2. Разработать бота с логикой работы из п.1 и возможностью общения пользователя с операторами, осуществляющими поддержку клиентов.
3. Подготовить сервер для обеспечения работы соответствующего ПО.

S - should, необходимые:

1. Обеспечить достаточное быстродействие бота. При том, что бот не относится к высоконагруженным.
2. Разместить ссылки на бота в социальных сетях для того, чтобы пользователи могли его найти и начать работу.
3. Проработать возможность и удобство создание личного кабинета абитуриентами.

C - could, нужные:

1. Проработать дизайн выдаваемой информации, чтобы выглядело не скучно, привлекательно. Сократить информацию, добавить ссылки, украсить сообщения эмодзи.
2. Сотрудники приёмной комиссии должны быть ознакомлены с суперчатом, создаваемым ботом, и понимать, как отвечать на вопросы пользователей
3. Разработать систему, выдающую статистическую информацию по работе бота.

W - would like, желаемые:

1. Добавить боту функцию рассылки сообщений.
2. Добавить функцию подбора оптимального направления для абитуриента.

**Функциональные требования:**

1. Чат-бот отправляет и обрабатывает кнопки двух видов.
2. Чат-бот может отправлять ссылки в тексте.
3. Чат-бот даёт возможность связаться с оператором.
4. Чат-бот дублирует информацию с сайта.
5. Чат-бот функционирует на имеющемся сервере.
6. Хранение базы вопросов и ответов осуществляется на имеющемся сервере.
7. Система должна иметь комплекс средств и мер обеспечения информационной безопасности, позволяя разграничивать права доступа пользователей к данным.
8. Платформа сохраняет всю историю обращений к Чат-боту.
9. Чат-бот реагирует на непредвиденные текстовые сообщения, возвращая пользователя в меню.
10. Чат-бот хранит информацию о пользователях в базах данных, позволяет пользователю изменить имеющиеся данные или дополнить их.

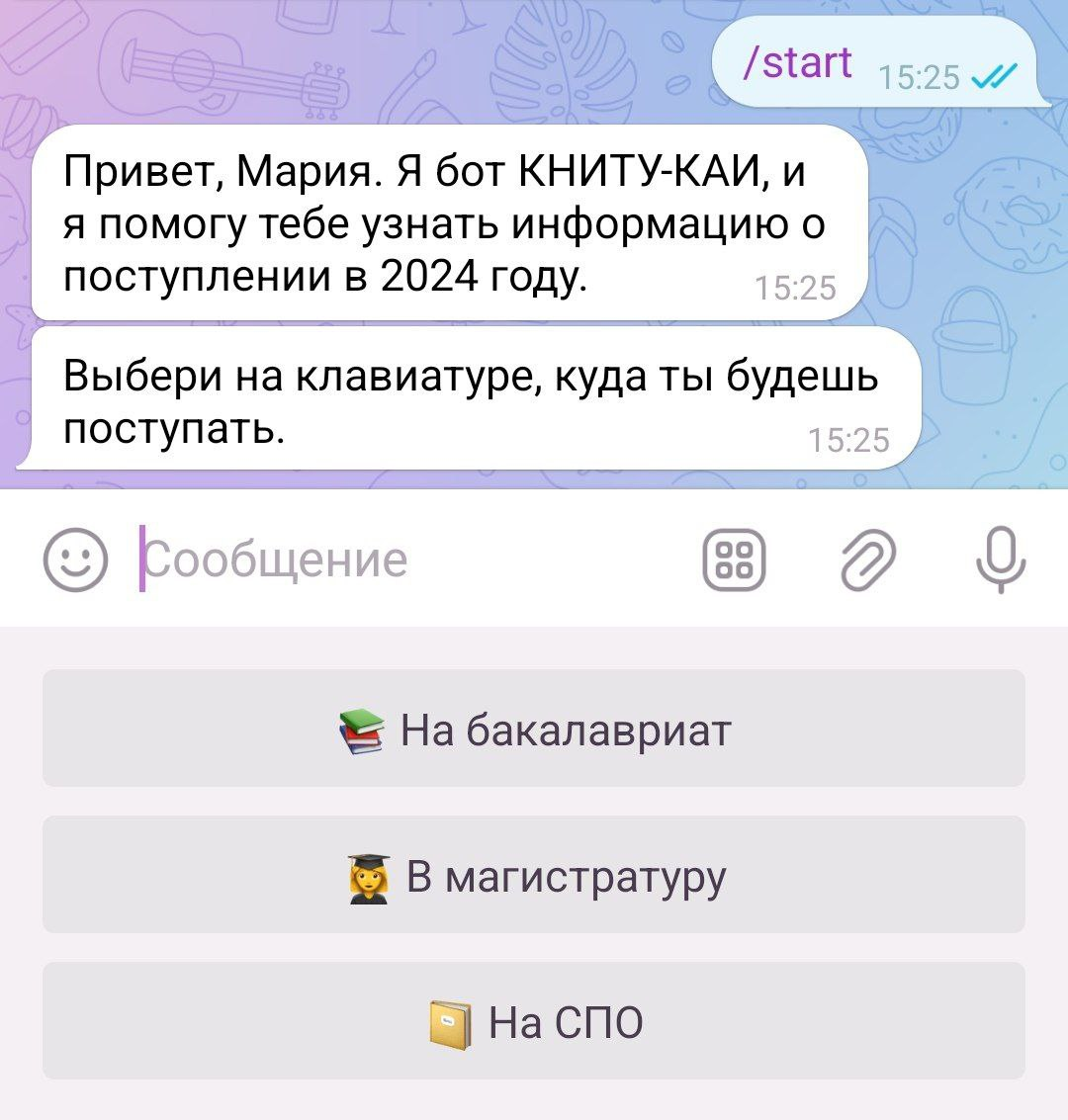
**Программная реализация проекта**

Важно провести тестирование перед запуском проекта. Показать бота наибольшему числу пользователей и убедиться в интуитивности интерфейса. При получении критики отредактировать бота для достижения наилучшего результата. В процессе эксплуатации также собирать информацию о работе бота и редактировать при необходимости.

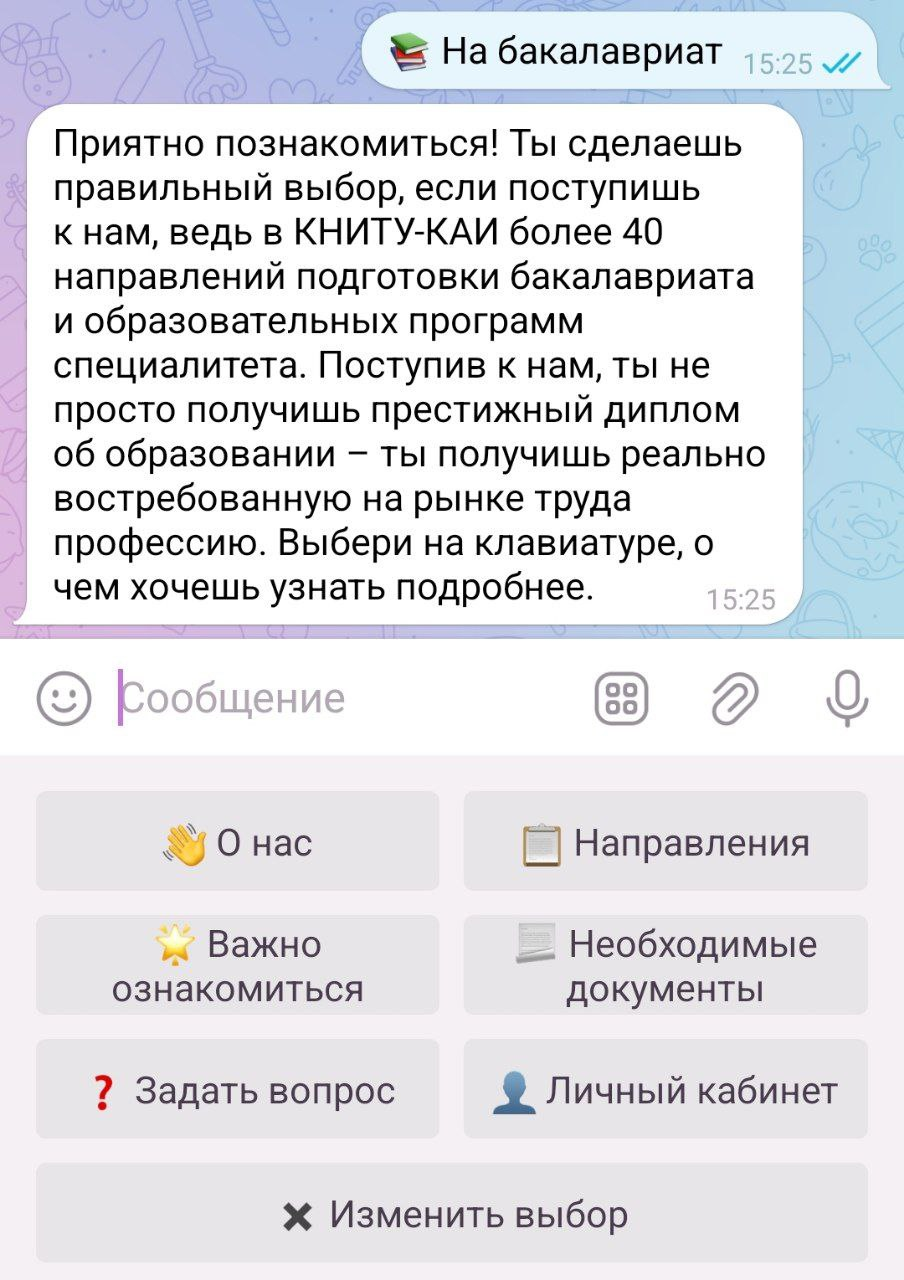
Языком программирования, используемым для разработки, сразу же был выбран Python, так как для этого языка существует несколько удобных и многофункциональных библиотек для создания Telegram-ботов. Наиболее удобной и актуальной оказалась библиотека aiogram 3. По данной версии библиотеки существуют подробные инструкции с примерами, что позволило найти ответ на имеющиеся вопросы.

Информация с сайта КНИТУ-КАИ была изучена, сгруппирована и сокращена, чтобы уместиться в формат текстового сообщения. На информацию, не поддающуюся сокращению, были оставлены ссылки.

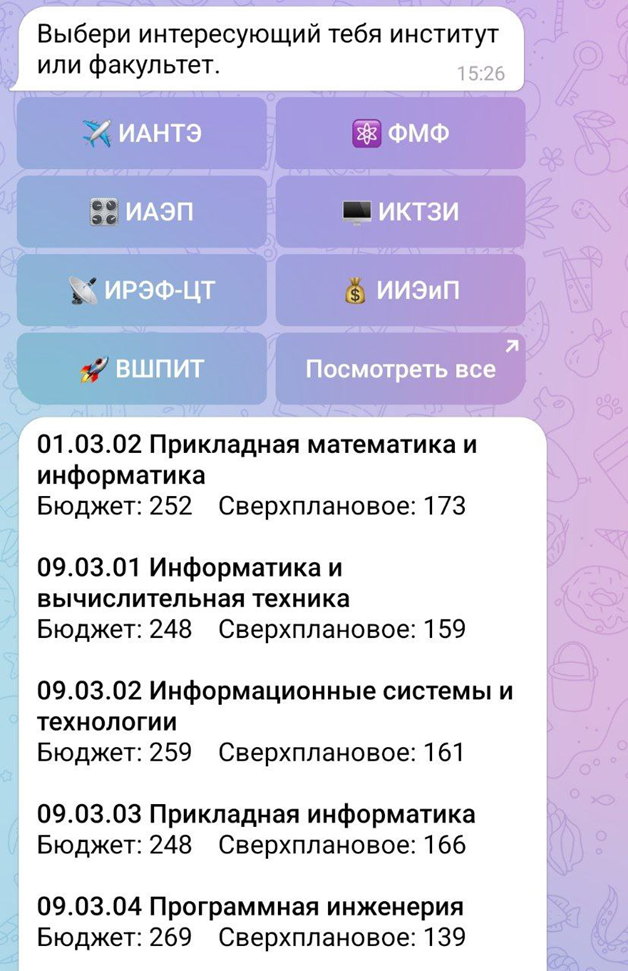
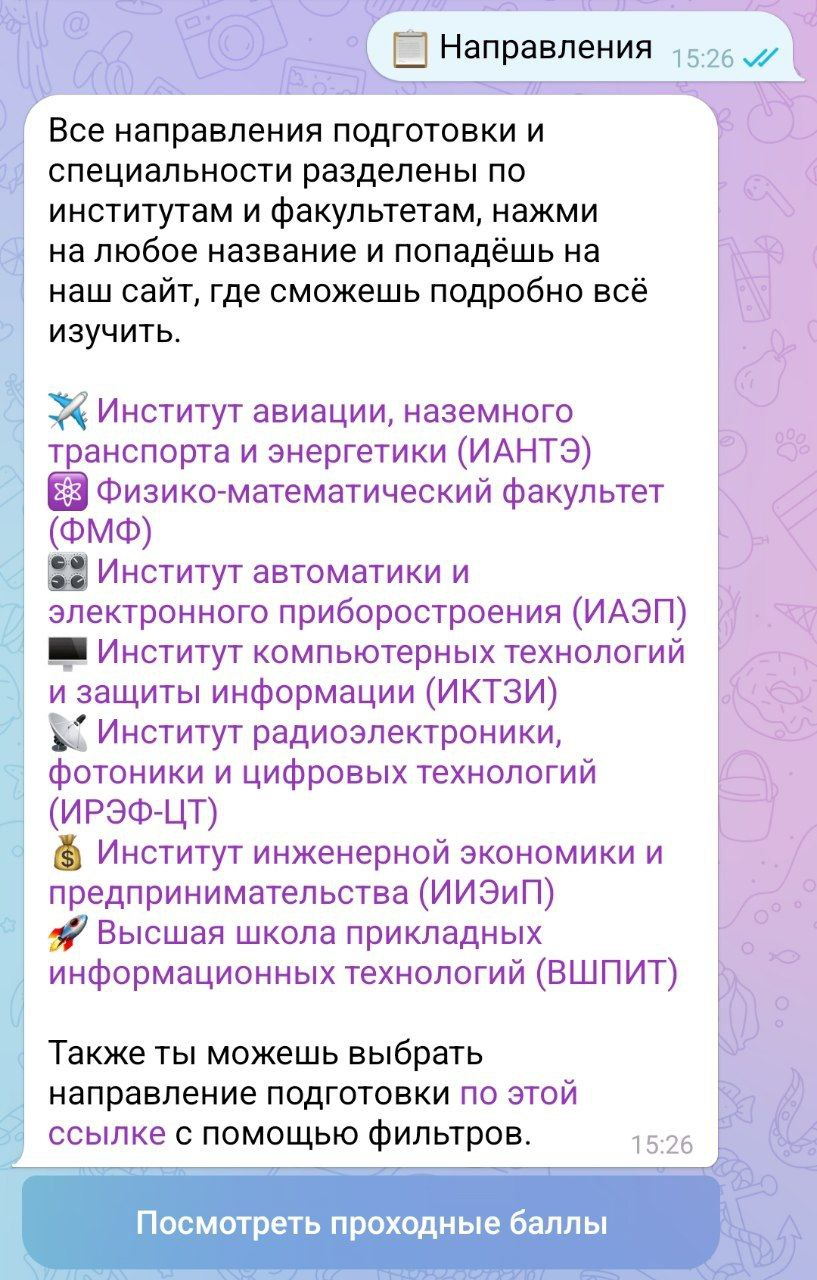
Изначальный интерфейс бота претерпел сильные изменения в процессе разработки. Первичный вид предоставлял возможность пользователям задать вопрос внутри бота с информацией. При этом механизм ответа для модераторов имел примитивную форму: не было возможности ознакомиться с прошлой перепиской, каждое сообщение представляло собой отдельный запрос. После первого тестирования было выявлено, что удобнее будет использовать формат аналогичный crm платформам. К тому же требовалось, чтобы ответ на сообщения также происходил внутри мессенджера Telegram. Был проведён поиск подходящего способа, найден проект Hotline CRM. Для работы платформы потребовалось создать отдельного бота для ответов на вопросы. Это позволило не только создать удобный интерфейс для модераторов, в котором сообщения от каждого пользователя группируются, но и предоставило пользователям возможность не потерять текст переписки с ответом в информации от первого бота.

  
Рис. 1. Первый запуск бота для пользователя.

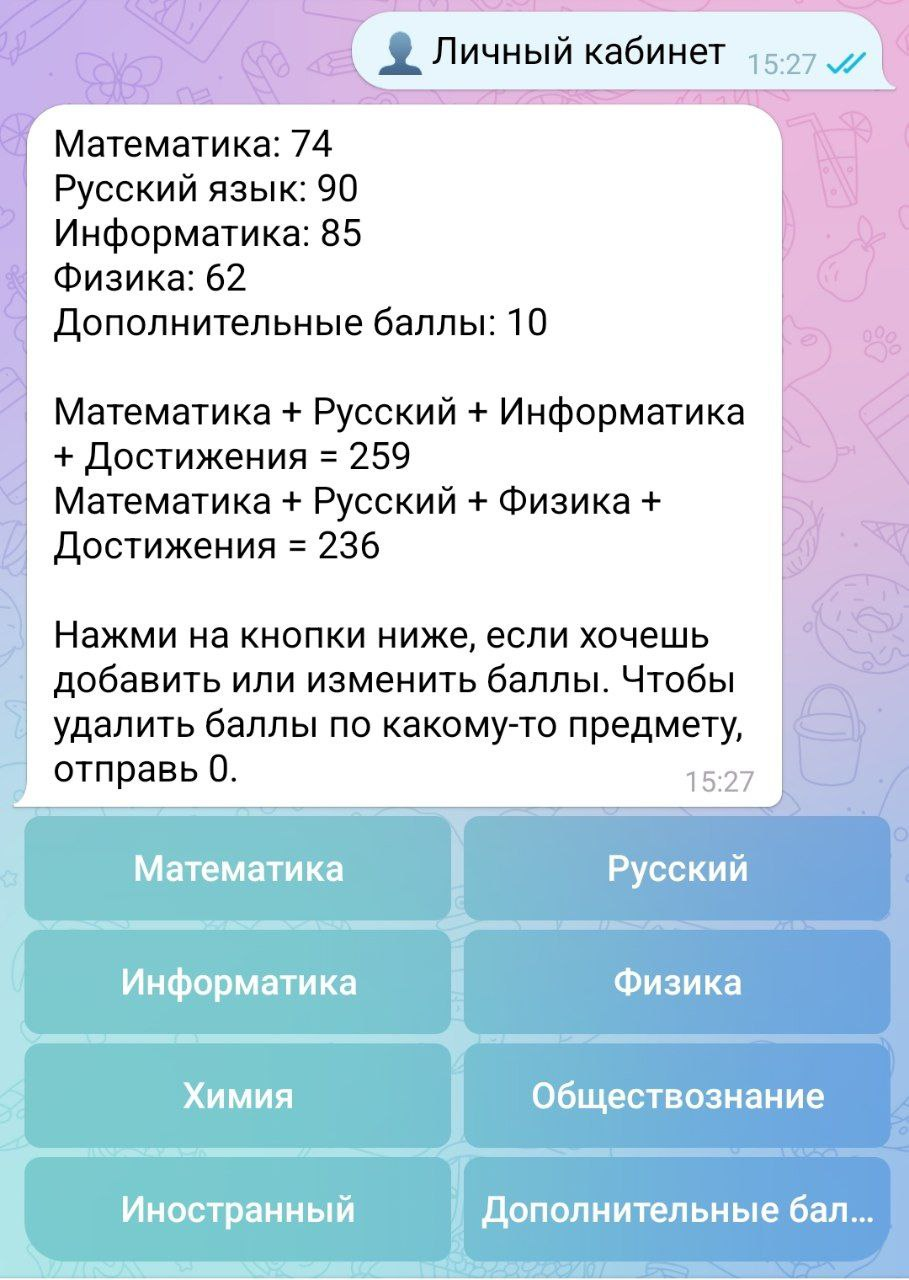
При запуске бота пользователь может выбрать, куда он хочет поступать: на бакалавриат, магистратуру или на среднее профессиональное образование (рис. 1). Для каждого выбора подготовлены отдельные текста с информацией о направлениях, университете. Информация о выборе пользователя хранится в базе данных. Пользователь в любой момент может изменить свой выбор на клавиатуре (рис. 2).

  
Рис. 2. Клавиатура для бакалавров и вступительный текст. Возможность изменения выбора.

Для бакалавров был добавлен удобный интерфейс просмотра проходных баллов, позволяющий выбрать интересующий институт или факультет и просмотреть проходные баллы прошлых лет, не выходя из Telegram (рис. 3, 4).

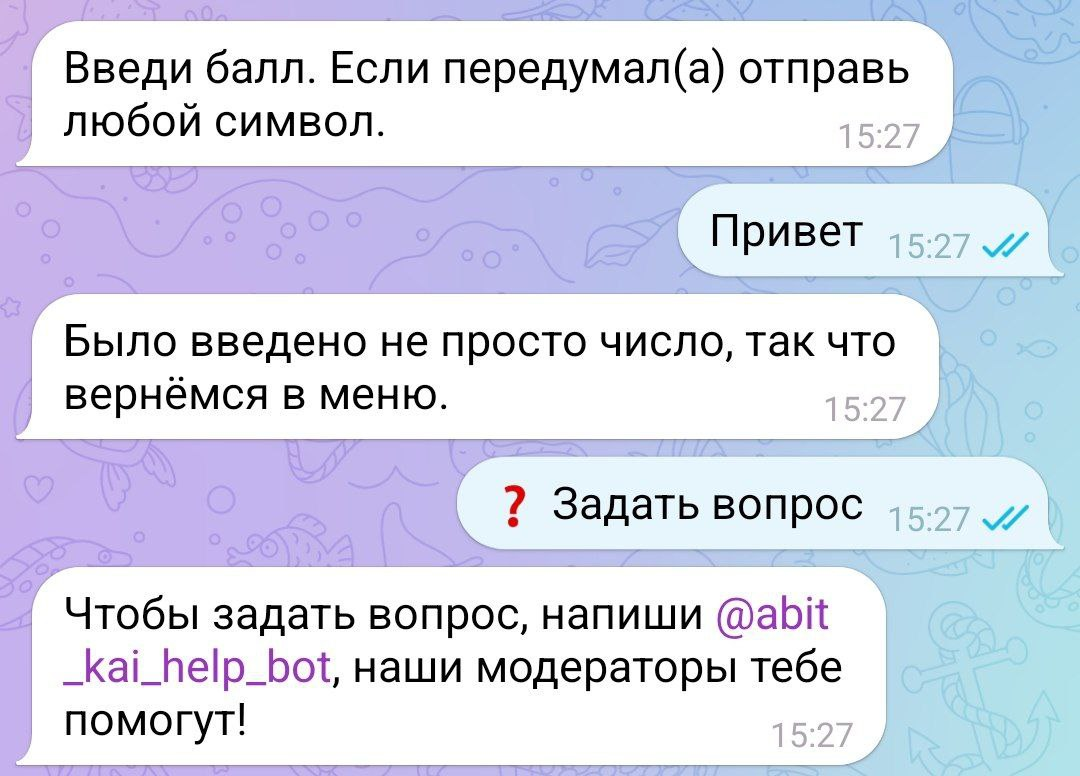
  
Рис. 3, 4. Просмотр проходных баллов прошлого года, разделенный по институтам и факультетам.

К тому же бакалаврам доступна функция «Личный кабинет». Пользователь может ввести свои баллы ЕГЭ и баллы за индивидуальные достижения, а в дальнейшем получить рекомендации по направлениям. В нём же доступен удобный подсчет баллов каждой комбинации предметов (Рис. 5). Так что, если пользователь сдавал больше трёх предметов, он получит доступ к сумме основных предметов с каждым из предметов по выбору, потому не будет вынужден утруждать себя подсчётом самостоятельно. В случае, если пользователь ещё не знает результаты некоторых экзаменов, он увидит сумму по имеющимся предметам.

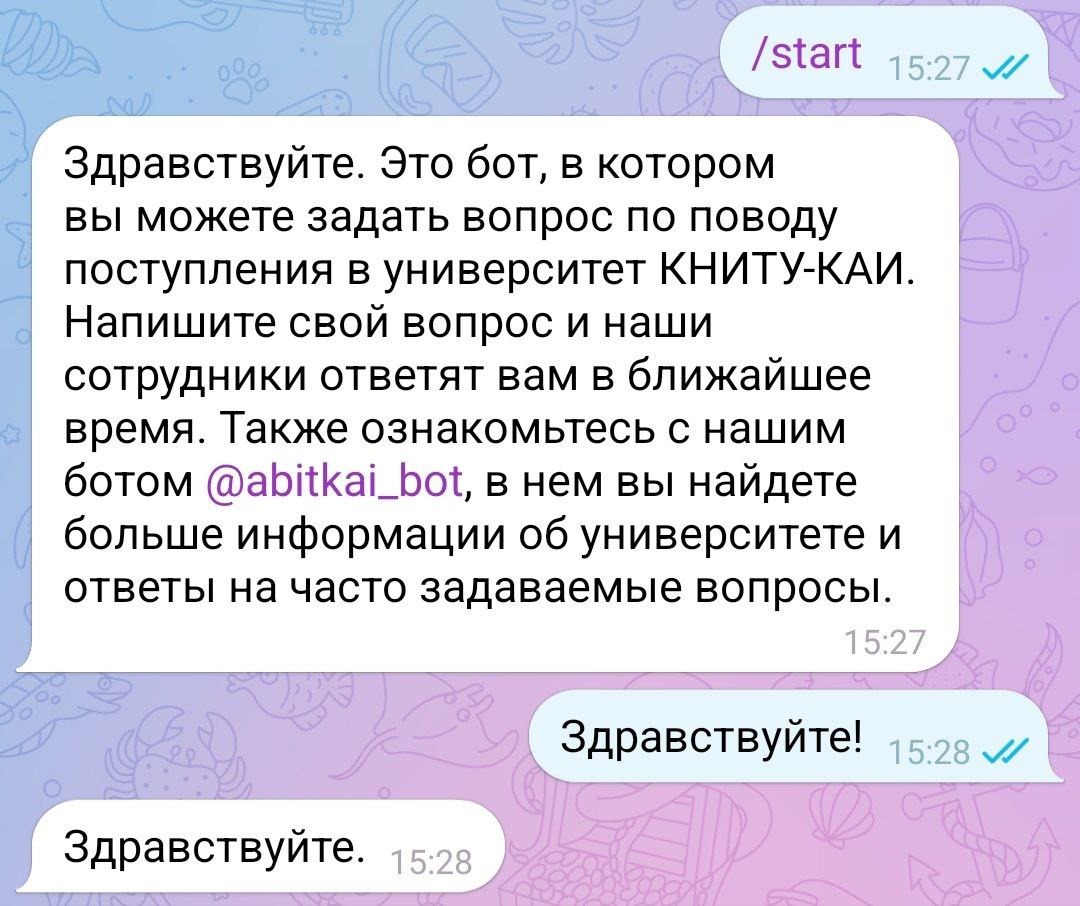
  
Рис. 5. Личный кабинет пользователя с возможностью добавить баллы по предметам и увидеть их сумму.

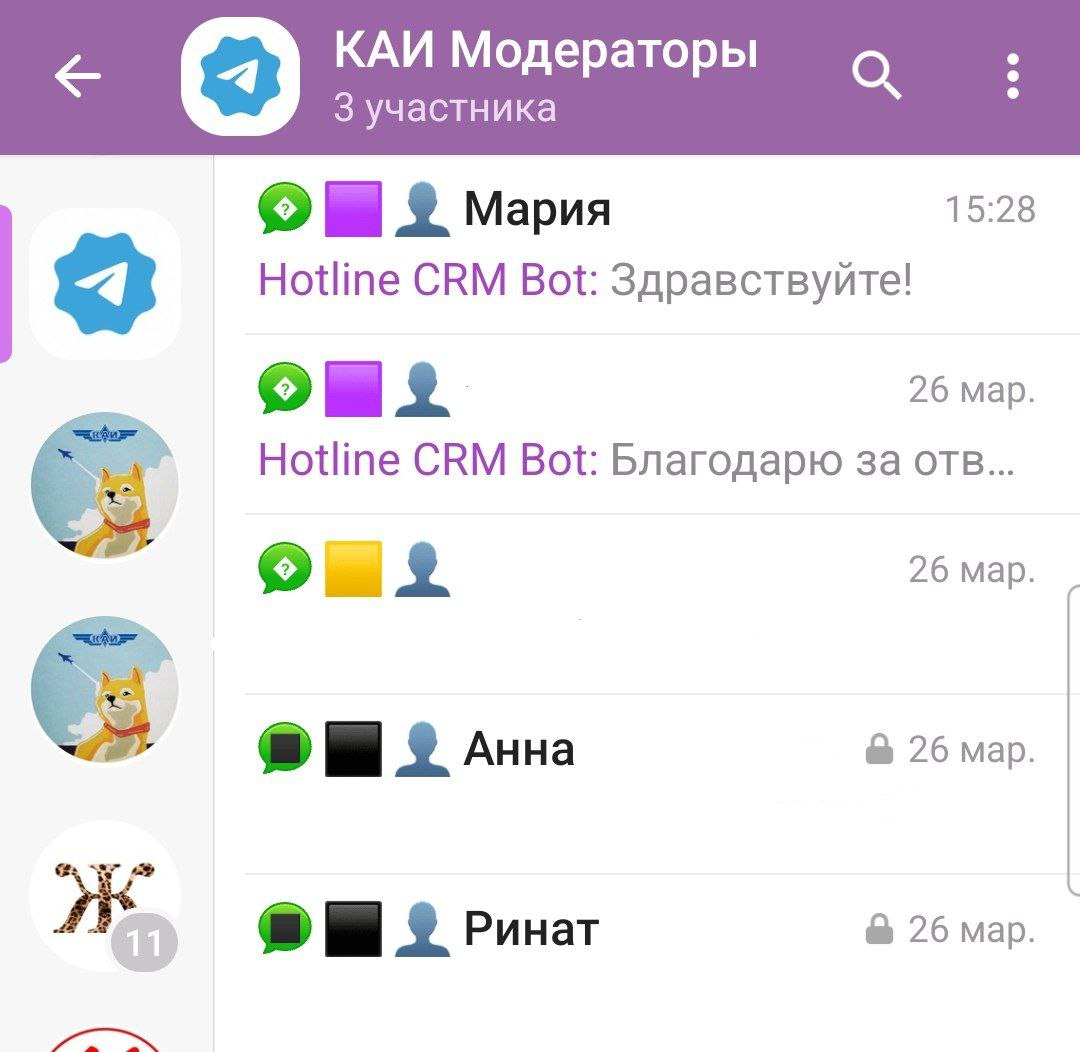
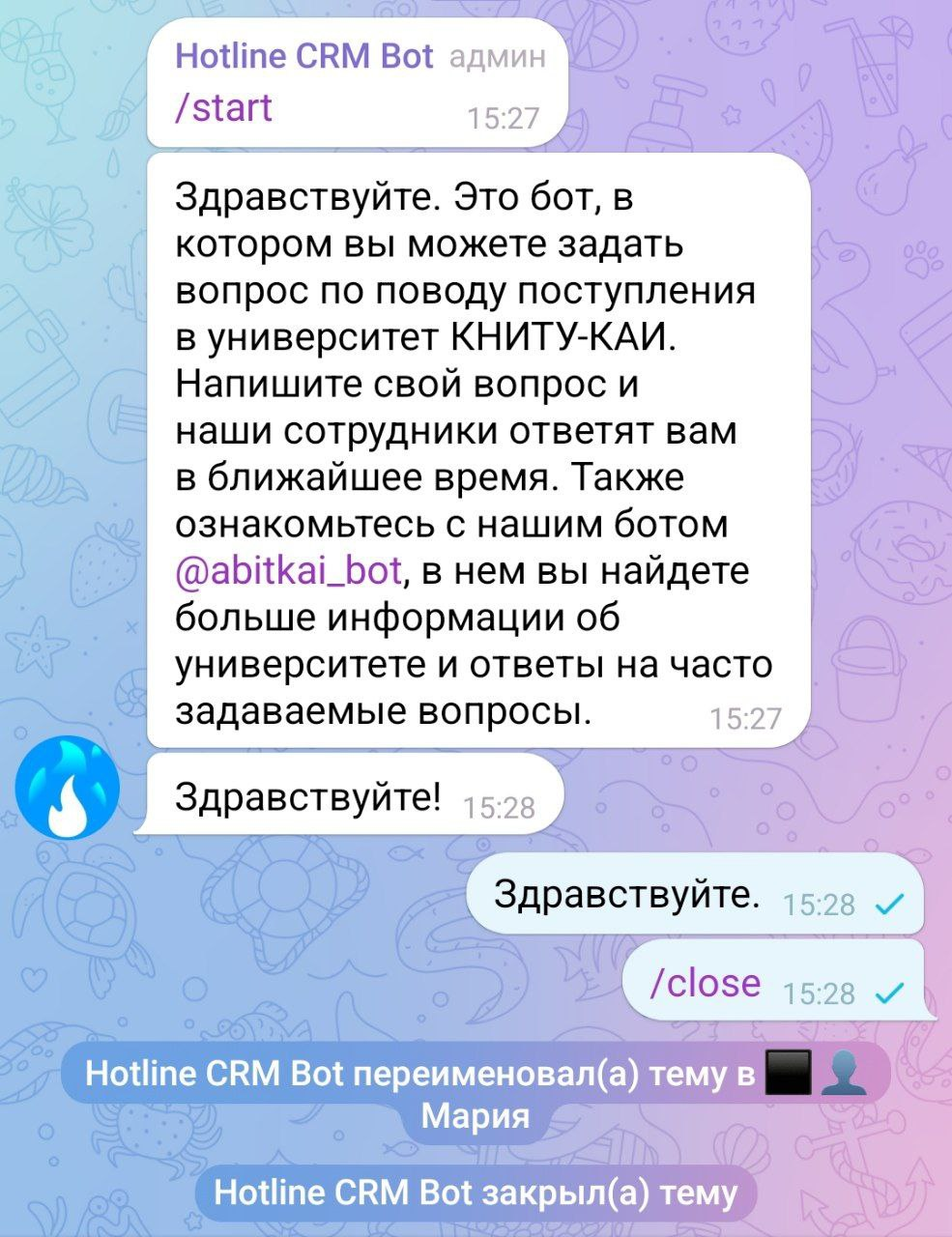
В бот была добавлена реакция на непредвиденные ответы. К тому же были опрошены первые пользователи и выявлены наиболее комфортные способы взаимодействия. Ввод команд оказался менее удобным, большинство пользователей предпочло использовать клавиатуру.

При нажатии кнопки «Задать вопрос» пользователю предлагаю перейти во второго бота, который позволяет связаться с модераторами (Рис. 6).

  
Рис. 6. Реакция на непредвиденный ответ и реакция на желание задать вопрос.

Когда пользователь переходит во второго бота, его встречает приветствие и предложение задать вопрос (Рис. 7). В суперчате, где находятся модераторы, создаётся новая тема, обозначенная фиолетовым цветом. Как только на вопрос было отвечено, цвет меняется на жёлтый, а затем тему можно закрыть, отправив команду /close, цвет сменится на чёрный (Рис. 8, 9). На вопрос пользователя может ответить любой модератор. Также у модераторов есть отдельная тема, где они могут обсудить, как ответить на вопросы.

  
Рис. 7. Интерфейс бота для ответов на вопросы для пользователя.

  
Рис. 8, 9. Интерфейс бота для ответов на вопросы для модераторов.

В качестве **механизмов реализации** был использован мессенджер Telegram, встроенный способ создания ботов, Python 3.9, библиотека aiogram 3, Hotline CRM bot.

**Ожидаемые результаты:** Telegram-бот реализован со всеми требуемыми функциями, запущен и беспрерывно работает все летние месяцы. Сотрудники приёмной комиссии ознакомлены с механизмом ответов на вопросы и считают его удобным. Осуществлено распространение информации о боте в социальных сетях университета, бот популярен у абитуриентов.

**Ключевые пользователи, производящие оценку результата:**

1. Абитуриенты. Целевая аудитория. Их мнение о боте крайне важно.
2. Сотрудники приёмной комиссии. Основные пользователи второго бота. Важно гарантировать им удобство ответов на вопросы.

**Достигнутые результаты:**

На данный момент результаты работы достигнуты частично: бот реализован и запущен на сервере. В летние месяцы эксплуатация бота будет контролироваться, ошибки будут исправляться, функциональность дополняться по запросу пользователей.

**Выводы:** В данной работе осуществлены: самостоятельная формулировка темы, посвящённая проекту по созданию Telegram-бота для абитуриентов, определены цели и задачи, критерии успеха, проблематика, заинтересованные стороны, сроки проекта, риски, метрики проекта, требования к продукту, функциональные требования, ожидаемые и достигнутые результаты, ключевые пользователи, производящие оценку. Выполнена разработка Telegram-бота и введение его в эксплуатацию.

Таким образом цель работы, заключающаяся в предоставлении возможности пользователям Telegram с помощью бота получать информацию о поступлении в университет, достигнута.

**Список литературы:**

1. Пишем Telegram-ботов с aiogram 3.x. URL: <https://mastergroosha.github.io/aiogram-3-guide/> <https://kai.ru> (дата обращения: 03.05.2024)
2. Hotline CRM. Офиц. Сайт. URL: <https://hotline.tg> (дата обращения: 03.05.2024)
3. Документация aoigram 3.5.0. Офиц. Сайт. URL: <https://docs.aiogram.dev/en/latest/> (дата обращения: 03.05.2024)
4. Документация языка программирования Python. Офиц. Сайт. URL: <https://www.python.org/doc/> (дата обращения: 03.05.2024)