



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Экономики и управления на предприятии природопользования»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(бакалаврская работа)
по направлению подготовки 09.03.03 Прикладная информатика
(квалификация – бакалавр)

На тему «Создание бота с применением технологий искусственного интеллекта»

Исполнитель Карагозян Алексей Сергеевич

Руководитель ст. преподаватель Сафонова Татьяна Владимировна

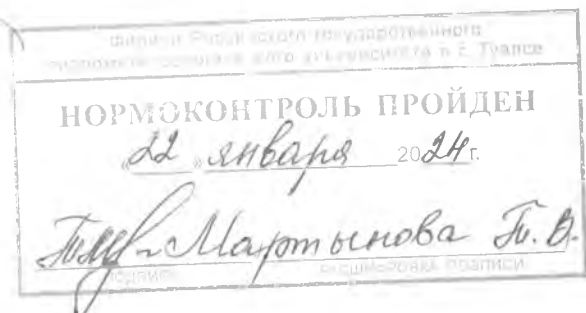
«К защите допускаю»

Руководитель кафедры

кандидат экономических наук

Майборода Евгений Викторович

«23» января 2024 г.



Туапсе
2024 г.

ОГЛАВЛЕНИЕ

Введение.....	3
1. Теоретические основы искусственного интеллекта.....	7
1.1 Понятие искусственного интеллекта и его применение	7
1.2 Основные методы и алгоритмы искусственного интеллекта.....	9
1.3 Обзор существующих решений в области создания ботов	12
2. Архитектура бота	17
2.1 Определение требований бота.....	17
2.2 Проектирование архитектуры бота.....	22
2.3 Выбор необходимых компонентов и технологий.....	25
2.4 Разработка и обучение модели искусственного интеллекта	27
2.5 Выбор и обоснование выбора алгоритма машинного обучения.....	30
2.6 Разработка и обучение модели на выбранных данных	32
3. Реализация и интеграция бота	35
3.1 Разработка бота с использованием выбранных технологий	35
3.2 Интеграция бота в выбранную платформу или систему.	42
3.3 Тестирование бота	44
Заключение	47
Список литературы	48

Введение

Создание ботов с применением технологий искусственного интеллекта — это инновационный подход, который позволяет программным системам имитировать интеллект и взаимодействовать с людьми на основе заданных алгоритмов и моделей. Искусственный интеллект обеспечивает ботам способность извлекать знания из больших объемов данных, анализировать информацию, обучаться и совершенствоваться со временем.

Боты с искусственным интеллектом широко применяются в различных сферах, включая бизнес-коммуникации, клиентское обслуживание, медицину, образование и многое другое. Они могут выполнять задачи, которые ранее могли быть выполнены только людьми, такие как отвечать на вопросы, предоставлять рекомендации, предупреждать о потенциальных проблемах и даже управлять другими устройствами.

Создание ботов с применением технологий искусственного интеллекта требует комплексного подхода, который включает в себя разработку основных алгоритмов, сбор и обработку данных, обучение моделей и реализацию системы в подходящей среде. Однако, благодаря возможностям искусственного интеллекта, создание ботов становится все более доступным и эффективным процессом.

В данной выпускной квалификационной работе мы рассмотрим основные этапы создания бота с использованием технологий искусственного интеллекта, а также представим примеры и практические рекомендации для разработки успешных ботов.

Продолжая о создании ботов с применением технологий искусственного интеллекта, стоит отметить, что их разработка требует использования различных подходов и инструментов. Для начала, необходимо определить цель создания бота и функциональность, которую он должен выполнять. Это может быть чат-бот для коммерческой поддержки или виртуальный помощник для обучения.

Следующим шагом является выбор подходящей технологии искусственного интеллекта. Существует широкий спектр технологий, включая нейронные сети, обработку естественного языка (Natural Language Processing, NLP), машинное обучение и глубокое обучение. Каждая из этих технологий может быть применена в разных аспектах разработки бота.

После выбора технологии следует собрать и подготовить данные для обучения бота. Это может включать сбор существующих данных или создание собственной базы данных для обучения модели. Необходимо убедиться, что данные достаточно разнообразны и репрезентативны для обучения бота.

Затем процесс обучения модели начинается. Здесь выбранные алгоритмы машинного обучения или глубокого обучения применяются к собранным данным для создания модели, способной отвечать на вопросы или выполнять требуемую функциональность. Обучение модели может занять некоторое время и требует итеративного процесса для настройки и улучшения ее работы.

Наконец, после завершения обучения модели, следует ее реализация и интеграция с выбранной платформой или средой. Это может быть настройка чат-интерфейса, создание веб-приложения или интеграция в существующую систему. Важно также создать механизмы оценки и отслеживания работы бота для постоянного совершенствования и улучшения его функциональности.

В итоге, создание ботов с применением технологий искусственного интеллекта открывает новые возможности для автоматизации и усовершенствования различных процессов и взаимодействий с клиентами. Благодаря постоянному развитию и продвижению искусственного интеллекта, боты становятся все более умными и эффективными, что делает их незаменимым инструментом для многих сфер деятельности [4].

Актуальность темы заключается в решении следующих проблем:

1. Оптимизация бизнес-процессов: создание бота с применением технологий искусственного интеллекта может помочь автоматизировать рутинные задачи и обработку информации. Это может существенно повысить эффективность работы организации и сократить затраты на персонал.

2. Улучшение клиентского опыта: боты, основанные на искусственном интеллекте, могут предоставлять быстрые и точные ответы на вопросы клиентов, помогая им получить нужную информацию или решить проблему без необходимости общения с реальным оператором. Это может улучшить уровень обслуживания и удовлетворенность клиентов.

3. Повышение конкурентоспособности: использование ботов на основе искусственного интеллекта становится все более популярным среди компаний разных отраслей. Внедрение таких технологий может помочь организации быть более конкурентоспособной на рынке и привлекать новых клиентов.

4. Анализ данных и прогнозирование: боты, основанные на искусственном интеллекте, могут обрабатывать большие объемы данных и анализировать их для выявления тенденций и предсказания будущих событий. Это может быть полезным инструментом для принятия бизнес-решений и улучшения стратегии компании.

5. Формирование персонального взаимодействия: боты могут адаптироваться под нужды и предпочтения каждого клиента, предлагая персонализированные рекомендации и услуги. Это может усилить взаимодействие с клиентами и повысить их лояльность к бренду.

Объект исследования:

1. Технологии искусственного интеллекта: исследование технологий, используемых для создания бота, включая машинное обучение.

2. Разработка алгоритмов: изучение различных алгоритмов, используемых для обучения и функционирования бота, например, алгоритмы глубокого обучения или рекуррентные нейронные сети.

3. Проектирование и разработка: исследование процесса проектирования и разработки бота, включая его архитектуру, функциональность, интерфейс и возможные способы взаимодействия с пользователем.

4. Обработка и понимание естественного языка: исследование методов и алгоритмов, которые позволяют боту понимать и обрабатывать естественный язык, включая распознавание речи, семантический анализ и генерацию текста.

5. Оценка производительности: исследование различных метрик и методов для оценки производительности и эффективности бота, включая точность ответов, скорость обработки запросов и масштабируемость.

6. Этические и юридические вопросы: рассмотрение этических и юридических аспектов создания и использования бота с применением технологий искусственного интеллекта, включая проблемы конфиденциальности данных и ответственность за действия бота.

Предмет исследования: разработка и оптимизация моделей искусственного интеллекта для создания бота с возможностью автоматизированных диалогов и определения пользовательских намерений.

Цель работы: целью данного исследования является создание эффективного бота с использованием технологий искусственного интеллекта, способного вести автоматизированные диалоги с пользователями, определить и интерпретировать их намерения, обеспечивая высокое качество и уровень удовлетворенности пользователей.

Задачи работы:

1. Изучение и анализ существующих технологий искусственного интеллекта, которые могут быть применены при создании бота.

2. Определение целей и функциональных возможностей бота.

3. Разработка архитектуры бота, включая модели обработки естественного языка, машинное обучение и алгоритмы принятия решений.

4. Сбор и подготовка данных для обучения бота, включая текстовые корпуса, размеченные данные или данные из внешних источников.

5. Обучение моделей на основе этих данных, используя алгоритмы машинного обучения, такие как нейронные сети или классические модели.

6. Разработка интерфейса пользователя для взаимодействия с ботом, например, текстового или голосового интерфейса.

7. Реализация алгоритмов обработки естественного языка, которые позволят боту понимать и генерировать текстовые ответы на запросы пользователя.

1. Теоретические основы искусственного интеллекта

1.1 Понятие искусственного интеллекта и его применение

Теоретические основы искусственного интеллекта (ИИ) являются фундаментальными принципами, концепциями и методами, лежащими в основе разработки и применения систем, способных проявлять интеллектуальное поведение и автоматически решать задачи, которые ранее считались прерогативой человека.

Одна из ключевых теоретических основ искусственного интеллекта — это теория символов. Она предполагает, что интеллектуальное поведение может быть описано как обработка символов, которые представляют знания и информацию. Символы представляются в компьютерной программе или модели, и манипулирование этими символами позволяет системе выполнять различные задачи.

Еще одной важной теоретической основой ИИ является теория вероятности и статистики. Она используется для моделирования неопределенности и нечеткости в данных и позволяет системе принимать решения на основе вероятностных выводов. Теория вероятности также применяется в статистическом обучении, где система использует данные для обучения и адаптации своих моделей и алгоритмов [13].

Машинное обучение является еще одной важной теоретической основой искусственного интеллекта. Оно представляет собой область, посвященную разработке алгоритмов и моделей, которые позволяют системе учиться на основе данных и опыта. Машинное обучение включает в себя различные методы, такие как нейронные сети, алгоритмы решения задач классификации и регрессии, кластеризации и многие другие.

Другая важная теоретическая основа ИИ — это теория решения задач. Она изучает методы и алгоритмы, позволяющие системе принимать решения и решать задачи эффективно. Множество методов решения задач было

разработано для различных сфер применения ИИ, включая поиск, планирование, оптимизацию и другие.

Наконец, область исследования знаний также является важной теоретической основой искусственного интеллекта. Она изучает способы представления и использования знаний в системе ИИ. Это включает в себя логические формализмы, алгоритмы вывода и рассуждения, методы анализа и извлечения знаний из данных.

Теоретические основы искусственного интеллекта предоставляют фреймворк и инструменты для разработки и применения интеллектуальных систем. Они изучают способы представления, обработки и использования информации и знаний, позволяя системам выявлять закономерности, принимать решения и решать задачи, схожие с теми, которые решает человек.

Кроме описанных выше теоретических основ, важным аспектом искусственного интеллекта является разработка алгоритмов и структур данных. Так, например, в области машинного обучения и глубокого обучения используются различные алгоритмы, такие как алгоритмы градиентного спуска, алгоритмы кластеризации, алгоритмы случайного леса и другие. Эти алгоритмы позволяют системам обучаться на основе данных, обнаруживать закономерности и принимать решения [4].

Структуры данных также играют важную роль в искусственном интеллекте. Например, для представления знаний и информации могут использоваться графы, деревья, матрицы и другие типы структур данных. Это позволяет эффективно организовывать и обрабатывать информацию в системе ИИ.

Другим важным аспектом теоретических основ искусственного интеллекта является формализация и моделирование знаний и представлений. Здесь используются логические формализмы, такие как исчисление предикатов или модальная логика, которые позволяют системам рассуждать, делать выводы и принимать решения на основе имеющихся знаний.

Еще одной важной теоретической основой искусственного интеллекта является теория игр. Она изучает стратегическое взаимодействие между агентами и оптимальные решения в ситуациях конфликта или сотрудничества. Теория игр находит применение в таких областях ИИ.

Наконец, этические и социальные аспекты искусственного интеллекта также имеют теоретические основы. Они изучают вопросы, связанные с ответственностью, прозрачностью и безопасностью систем ИИ, а также вопросы, связанные с влиянием ИИ на общество и человеческое поведение.

Все эти теоретические основы совместно образуют базу для разработки и применения искусственного интеллекта. Их изучение позволяет улучшить алгоритмы, разработать новые модели и методы, а также эффективно применять ИИ в различных сферах, включая медицину, финансы, автоматизацию процессов и другие [19].

1.2 Основные методы и алгоритмы искусственного интеллекта

1. Методы машинного обучения:

1) Решающие деревья (Decision trees): используются для классификации объектов, основываясь на их характеристиках и правилах принятия решений.

2) Нейронные сети (Neural networks): моделируют работу нервной системы человека, способны обучаться и адаптироваться к новым данным.

3) Кластеризация: группировка объектов по их сходству на основе расстояний между признаками.

4) Метод опорных векторов (Support Vector Machines): используется для классификации, основываясь на построении разделяющей гиперплоскости между объектами разных классов.

5) Генетические алгоритмы: воспроизводство и эволюция генетическим путем для поиска оптимальных решений.

2. Эвристические методы:

1) Экспертные системы: базируются на знаниях и опыте экспертов в предметной области и используют ЕС-правила для решения задач.

2) Генетическое программирование: эволюция программ путем мутации и скрещивания, чтобы найти оптимальное решение.

3) Распознавание образов: алгоритмы, позволяющие идентифицировать и классифицировать образы на основе их характеристик.

3. Обучение с подкреплением:

1) Алгоритм Q-обучения: агент обучается на основе опыта взаимодействия с окружающей средой и награды, стремясь максимизировать получаемый выигрыш.

2) Марковские процессы принятия решений (MarkovDecisionProcesses): модель, учитывающая принятие последовательных решений с учетом вероятности и награды.

4. Естественно-языковые методы:

1) Обработка текста: алгоритмы анализируют и интерпретируют текстовую информацию для извлечения смысла.

2) Машинный перевод: перевод текстов с одного естественного языка на другой с использованием алгоритмов искусственного интеллекта.

3) Голосовые ассистенты: системы, способные обрабатывать и отвечать на голосовые команды и запросы пользователей.

5. Глубокое обучение (Deep learning):

1) Использует искусственные нейронные сети с большим количеством слоев для обработки и анализа данных.

2) Применяется в различных задачах, таких как распознавание и классификация изображений, анализ естественного языка, обработка речевых сигналов и др.

3) Имеет возможность извлечения высокоуровневых признаков из данных и автоматического обучения без необходимости предварительного определения признаков.

6. Рекуррентные нейронные сети (Recurrent Neural Networks, RNN):

1) Используются для обработки и анализа последовательностей данных, таких как временные ряды, текстовые данные, речь и др.

2) Могут сохранять информацию о предыдущих состояниях сети, что позволяет учитывать контекст и зависимости в последовательности.

7. Метод оптимизации:

1) Градиентный спуск (GradientDescent): используется для настройки параметров модели путем минимизации функции потерь.

2) Стохастический градиентный спуск (StochasticGradientDescent): вариант градиентного спуска, который обновляет параметры модели на каждом шаге обучения, используя только часть обучающей выборки.

3) Adam: расширение градиентного метода оптимизации, учитывающее моменты первого и второго порядка.

8. Эволюционные алгоритмы:

1) Генетическое программирование: эволюция программ, в том числе искусственных нейронных сетей, путем мутации и скрещивания.

2) Генетический алгоритм: применение механизмов естественного отбора, мутации и скрещивания для поиска оптимальных решений.

9. Обработка изображений и компьютерное зрение:

1) Сверточные нейронные сети (ConvolutionalNeural Networks, CNN): специализированные для анализа и классификации изображений, используют операцию свертки для извлечения локальных признаков.

2) Алгоритмы сегментации изображений: разделение изображений на составные части для дальнейшего анализа.

3) Детектирование объектов: определение и локализация объектов на изображениях с использованием методов машинного обучения.

10. Обработка естественного языка (Natural Language Processing, NLP):

1) Алгоритмы обработки текста и анализа смысла: определение темы текста, извлечение сущностей, классификация тональности и др.

2) Методы машинного перевода: автоматический перевод текстов с одного языка на другой.

3) Генерация текста: создание текстовых данных на основе заданных правил и контекста.

Это лишь некоторые из основных методов и алгоритмов искусственного интеллекта, которые используются в научных исследованиях и разработке AI-систем. Также важно отметить, что эти методы непрерывно развиваются и совершенствуются вместе с появлением новых технологий и данных[6].

1.3 Обзор существующих решений в области создания ботов

Можно выделить несколько существующих решений в области создания ботов:

1. Dialogflow:

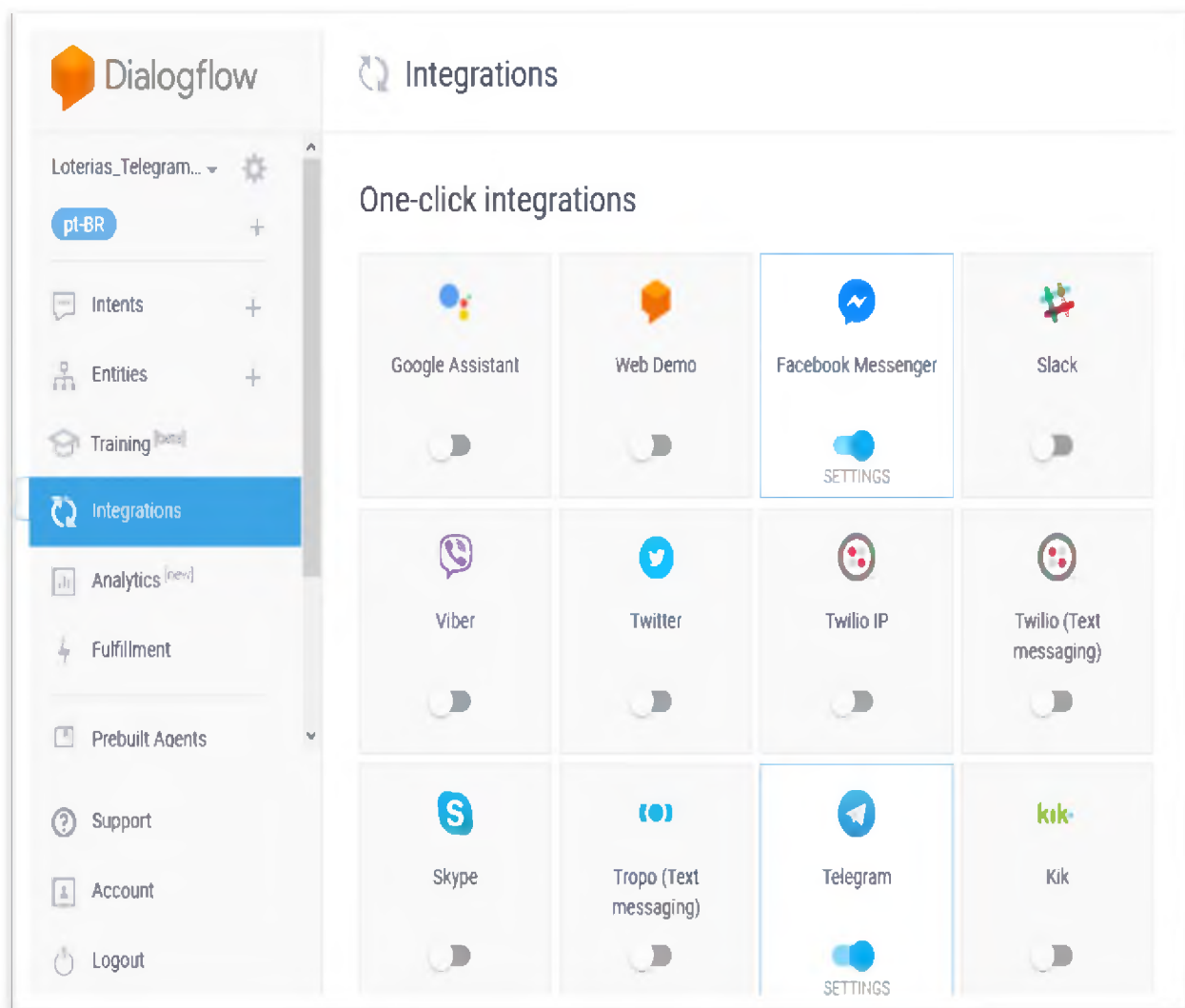


Рисунок 1 – Функциональность Dialogflow.

Dialogflow – это партнер Google Cloud для создания чат-ботов с использованием естественного языка [3].

- Поддерживает интеграцию с различными платформами, включая Facebook Messenger, Slack, WhatsApp и др. (рисунок 1).

- Предоставляет возможности распознавания и анализа естественного языка, создания ответов и управления диалогом с помощью логических блоков.

- Имеет функции машинного обучения для постоянного улучшения работы бота.

2. WatsonAssistant:

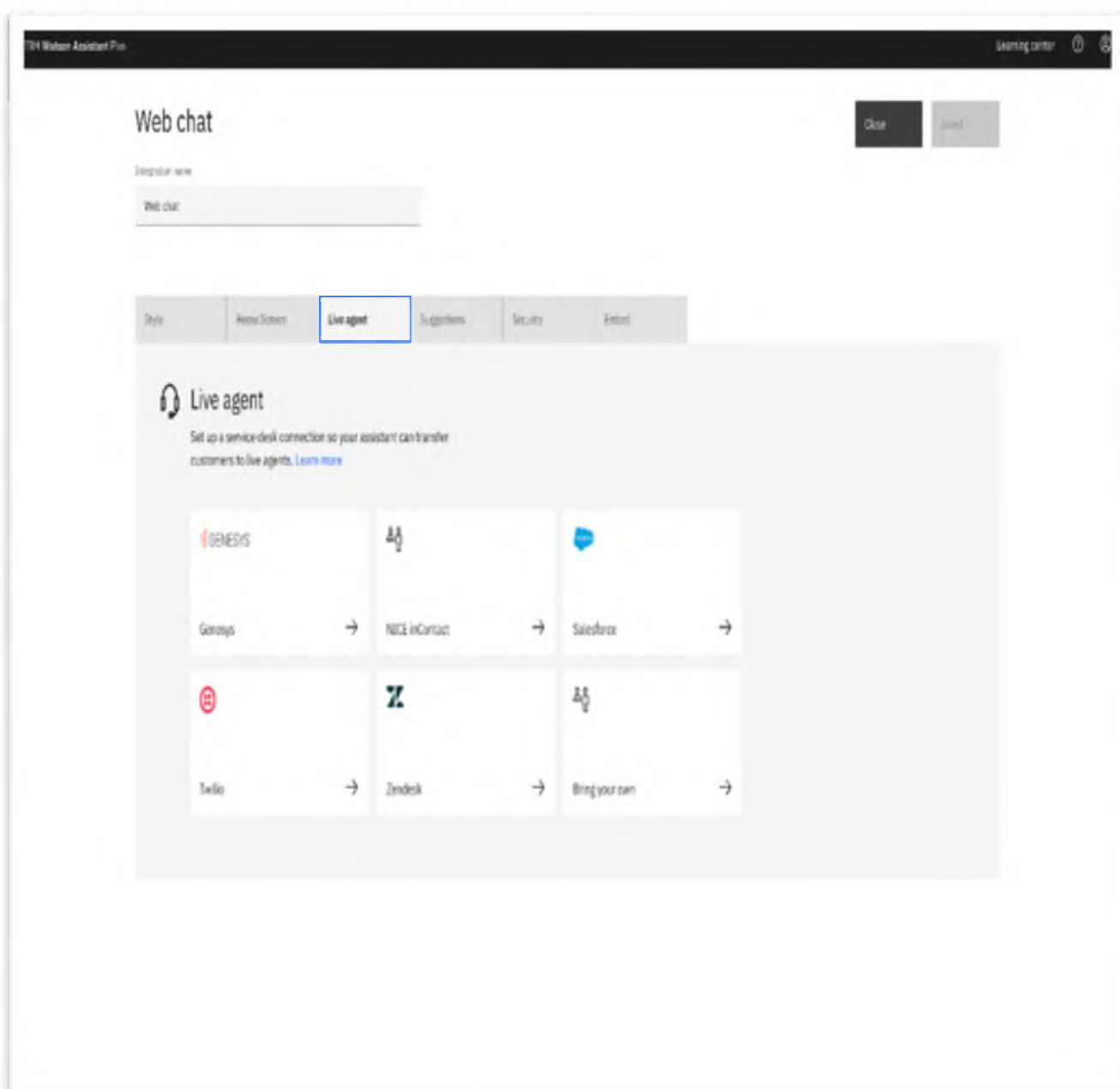


Рисунок 2 – Возможности WatsonAssistant.

- От IBM, предоставляет возможности по созданию и управлению различными типами ботов.
- Позволяет разработчикам создавать чат-ботов с помощью визуального интерфейса.
- Поддерживает множество языков и платформ, включая Facebook Messenger, Slack, Telegram и др.
- Имеет возможность обучения и адаптации на основе данных пользователей[4][10].

3. Amazon Lex:

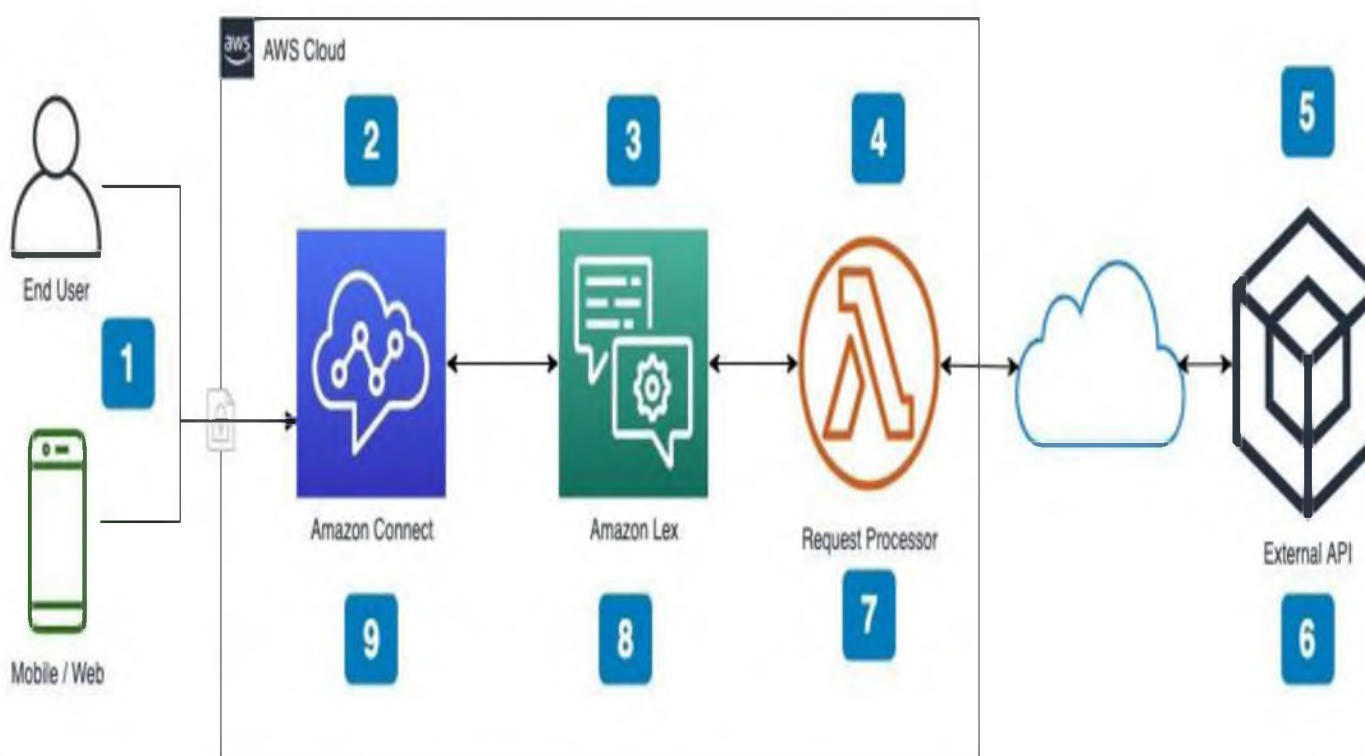


Рисунок 3 – Интеграция Amazon Lex

- Сервис от Amazon Web Services для создания ботов с использованием голосовых и текстовых интерфейсов.
- Имеет интеграцию с платформами Amazon Alexa и Slack.
- Позволяет создавать модели разговоров и управлять диалогами с помощью слотов и индентов.
- Обеспечивает масштабируемость и надежность благодаря инфраструктуре AWS[1].

4. Microsoft Bot Framework:

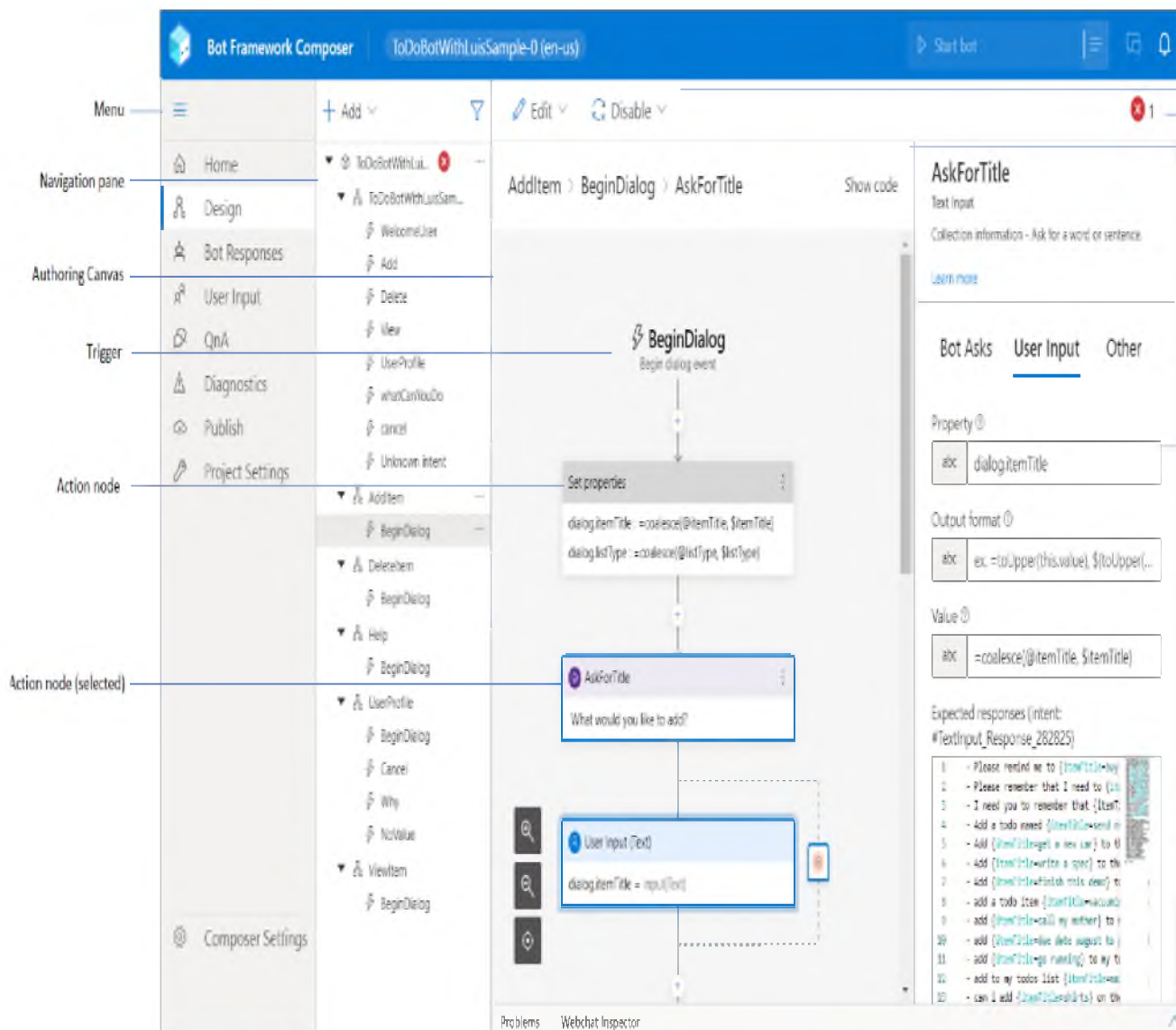


Рисунок 4 – Простой интерфейс Microsoft Bot Framework.

Инструментарий от Microsoft для разработки и развертывания ботов на различных платформах:

- Поддерживает разработку ботов с использованием языка C# или JavaScript.
- Имеет интеграцию с платформами Microsoft Teams, Skype, Slack и др.
- Обладает возможностью обработки естественного языка и создания диалогов с помощью библиотеки BotBuilder [6].

5. ManyChat:

- Сервис для создания ботов в мессенджерах, включая Facebook

Messenger.



Рисунок 5 – ManyChat.

- Имеет визуальный конструктор для создания бота без необходимости программирования.
- Обладает функционалом для создания последовательного потока сообщений и взаимодействия с пользователями.
- Предоставляет инструменты для маркетинга и продвижения бота [2].

Каждое из указанных решений имеет свои особенности и преимущества, в зависимости от требований и целей создания бота. При выборе подходящего решения рекомендуется учитывать функционал, интеграцию с платформами, возможности машинного обучения, гибкость настройки диалогов и другие важные факторы [21].

2. Архитектура бота

2.1 Определение требований бота

Архитектура бота определяет структуру и организацию компонентов бота для обеспечения его функциональности и эффективности. Вот несколько основных компонентов, которые могут входить в архитектуру бота:

1. Интерфейс пользователя:

- Определяет, как пользователи будут взаимодействовать с ботом, например, через мессенджеры, голосовые помощники или веб-сайты.

- Включает в себя дизайн и разработку пользовательского интерфейса для удобства использования и понимания бота.

2. Обработка естественного языка (Natural Language Processing, NLP):

- Распознавание и понимание введенных пользователем команд и запросов.

- Включает в себя компоненты, такие как различение интенгов (цели) и сущностей (параметров команды), обработка языковых конструкций и анализ тональности.

3. Управление диалогом:

- Механизм, который определяет структуру и поток диалога с ботом.

- Включает в себя определение логики ответов бота, контекста, гибкости в обработке вопросов и команд пользователя.

4. Бизнес-логика:

- Логика, связанная с конкретными требованиями и целями бота.

- Определяет, как бот будет обрабатывать и выполнять запросы пользователей, какие операции выполнять и как предоставлять информацию или решать проблемы.

5. Интеграция с сервисами и базами данных:

- Включает в себя интеграцию с различными внешними сервисами и базами данных для получения и обработки данных, доступ к API и другими необходимыми ресурсами [16].



Рисунок 6 – SWOT-анализ бота

SWOT-анализ, показанный на рисунке 6 (анализ преимуществ, слабостей, возможностей и угроз) для бота искусственного интеллекта выглядит следующим образом:

Преимущества:

1. Высокая скорость и эффективность обработки информации. Боты искусственного интеллекта могут быстро анализировать и обрабатывать большой объем данных, что позволяет решать задачи быстрее, чем человек.

2. Способность к обучению и адаптации. Боты могут улучшать свою работу по мере взаимодействия с пользователями и основываясь на своем опыте.

3. 24/7 доступность. Боты могут работать круглосуточно без необходимости отдыха или перерывов.

4. Масштабируемость. Боты могут обслуживать большое количество пользователей одновременно без потери качества обслуживания.

Слабости:

1. Ограниченность в области эмоционального интеллекта. Боты искусственного интеллекта могут затрудняться в распознавании и адекватной реакции на эмоции людей, что может отрицательно влиять на качество взаимодействия и удовлетворенность пользователей.

2. Недостаточность в понимании контекста. Боты могут терять связь с предыдущими разговорами или не уметь интерпретировать сложные вопросы, требующие глубокого понимания контекста.

3. Ограниченность в креативности и интуиции. Боты обычно ориентированы на заданные алгоритмы и данные, что ограничивает их способность к креативным решениям или интуитивным выводам.

Возможности:

1. Расширение функционала. Боты искусственного интеллекта могут быть использованы в различных областях, как коммуникационные помощники, персональные помощники, в автоматизации процессов и т.д.

2. Развитие и улучшение алгоритмов машинного обучения. Продолжающиеся исследования и разработки могут привести к созданию более умных и интуитивных ботов.

3. Интеграция с другими технологиями. Боты могут быть интегрированы с другими системами и технологиями, такими как голосовые помощники или робототехника.

Угрозы:

1. Уязвимость для взломов. Боты искусственного интеллекта могут подвергаться кибератакам или вредоносным действиям злоумышленников.

2. Негативное воздействие на трудовые рынки. Автоматизация задач, которые ранее выполняли люди, может привести к сокращению рабочих мест и негативно сказаться на определенных отраслях.

3. Потенциальное несоответствие этическим нормам. Боты искусственного интеллекта могут принимать решения, которые противоречат этическим стандартам или вызывают недоверие у пользователей.

В общем, SWOT-анализ позволяет определить преимущества, слабости,

возможности и возможные угрозы, связанные с использованием ботов искусственного интеллекта, что помогает разрабатывать стратегии и принимать решения, связанные с их использованием.

Определение требований к боту — это процесс определения целей, функциональных и нефункциональных требований, которые должен удовлетворять бот. Вот некоторые основные аспекты, которые нужно учитывать при определении требований к боту:

1. Цели и задачи:

- Четкое определение того, для чего будет использоваться бот (например, обслуживание клиентов, предоставление информации, выполнение операций и т. Д.).

- Определение ключевых функций и возможностей бота для достижения поставленных целей.

2. Виды взаимодействия:

- Определение типов взаимодействия, которые должен поддерживать бот (текст, голос, изображения), а также интеграцию со специфическими платформами и каналами связи.

3. Функциональные требования:

- Описание основных функций и возможностей бота, таких как обработка команд и запросов пользователей, генерация ответов, выполнение операций, интеграция с другими системами и т. Д.

4. Пользовательский опыт:

- Учет требований к удобству использования и пониманию пользовательского интерфейса, проектирование диалогов и логики ответов, чтобы обеспечить положительный опыт пользователя.

5. Безопасность и конфиденциальность:

- Разработка мер безопасности для обеспечения защиты данных пользователей, обеспечение конфиденциальности и безопасности информации.

6. Производительность и масштабируемость:

- Обеспечение эффективности работы бота, его способности к обработке

большого количества запросов, поддержка масштабируемости и высокой производительности.

7. Требования к интеграции с другими системами:

- Определение необходимости интеграции бота с другими внешними сервисами, базами данных или API для доступа и обработки данных.

Учет этих и других факторов поможет определить требования к создаваемому боту и обеспечить его соответствие целям и ожиданиям пользователей.

8. Поддержка многопользовательской среды:

- Обеспечение возможности одновременного взаимодействия с ботом несколькими пользователями.

- Реализация функционала по работе с сессиями и учета контекста диалога для каждого пользователя.

9. Адаптивность и многоязыковая поддержка:

- Учет требований к адаптивности бота под разные типы устройств и платформ.

- Поддержка нескольких языков для обеспечения доступности и удобства использования различными пользователями.

10. Мониторинг и аналитика:

- Возможность отслеживания работы бота, сбора и анализа данных о взаимодействии пользователей, выявление слабых мест и улучшение функционала.

- Интеграция с системами мониторинга и аналитики для непрерывного улучшения работы бота.

11. Обучение и адаптация:

- Возможность обучения и обновления бота на основе опыта взаимодействия с пользователями.

- Предусмотренные механизмы для улучшения работы бота на основе новых данных и обратной связи пользователей.

12. Способность к интеграции с машинным обучением и алгоритмами

искусственного интеллекта:

- Возможность использования алгоритмов машинного обучения и AI для улучшения диалогового опыта и предоставления более точных ответов пользователю.

13. Расширяемость и гибкость:

- Предусмотренные возможности добавления новых функций и возможностей, настройка и изменение логики работы бота без необходимости переработки всей архитектуры.

14. Учет требований к защите данных:

- Разработка мер по обеспечению безопасности данных пользователей, защите данных и соблюдению приватности, с учетом соответствующих стандартов и законодательства.

Определение требований представляет собой ключевой этап в разработке бота, который помогает успешно спроектировать его архитектуру, функциональность и возможности. У чтение указанных требований позволяет создать бота, который будет соответствовать ожиданиям пользователей и целям его использования [27].

2.2 Проектирование архитектуры бота

Проектирование архитектуры бота состоит из нескольких этапов и включает в себя определение функциональных требований, выбор технологий, определение архитектурных компонентов и связей между ними, а также проектирование пользовательского интерфейса.

Первым шагом при проектировании архитектуры бота является изучение функциональных требований. Здесь необходимо определить, какие задачи должен выполнять бот, какие данные он должен обрабатывать и какие функции должен предоставлять пользователю.

После этого следует выбор технологий, которые будут использоваться при разработке бота. Это может быть выбор конкретной платформы для

разработки (например, Facebook Messenger или Telegram), а также выбор языка программирования и инструментов разработки.

Далее необходимо определить основные компоненты архитектуры бота и связи между ними. Компоненты могут включать в себя модуль обработки естественного языка (Natural Language Processing), модуль структурирования данных, модуль авторизации и др.

Важно продумать, какие данные и какой функционал будет храниться в каждом компоненте и как они будут взаимодействовать между собой.

Также стоит уделить внимание проектированию пользовательского интерфейса. Это может быть как текстовый интерфейс, так и голосовой или графический.

Важно определить, какие функции и данные будут доступны пользователю, а также организовать навигацию и взаимодействие с ботом.

В процессе проектирования архитектуры бота также стоит учитывать масштабируемость системы, чтобы бот мог обрабатывать большое количество запросов и быть готовым к дальнейшему развитию и добавлению нового функционала [5].

После определения функциональных требований, выбора технологий и компонентов архитектуры бота, следует более подробное проектирование каждого компонента.

Модуль обработки естественного языка (NLP) является одним из ключевых компонентов архитектуры бота. Он отвечает за анализ и понимание вводимых пользователем текстовых данных.

Для этого может использоваться алгоритмы машинного обучения, обработка естественного языка (NLP) или комбинация различных методов. Важно продумать, какие языковые модели будут использоваться и каким образом они будут обучаться и обновляться.

Следующим компонентом является модуль структурирования и обработки данных. Он отвечает за получение, хранение и обработку данных, предоставляемых пользователю.

Важно определить, какие типы данных будут использоваться (текстовые, графические, голосовые и т. Д.), а также организовать их хранение и доступ.

Модуль авторизации и безопасности – ещё один важный компонент. Он отвечает за аутентификацию пользователей, проверку доступа к определенным функциям бота, защиту данных и обеспечение конфиденциальности. Здесь важно продумать методы аутентификации (логин/пароль, OAuth и т. Д.) и механизмы шифрования данных.

Дополнительные компоненты архитектуры бота могут включать интеграцию с другими системами (например, CRM или базой данных), модуль аналитики и отчетности для мониторинга работы бота и сбора статистики, а также модуль управления и обновления бота [9].

После проектирования компонентов и их взаимосвязей, стоит провести тестирование архитектуры бота для проверки его работоспособности и соответствия требованиям.

Это может включать юнит-тестирование каждого компонента, интеграционное тестирование, функциональное тестирование и тестирование производительности.

В итоге, проектирование архитектуры бота – это комплексная задача, включающая не только определение функциональных требований и выбор технологий, но и организацию компонентов и связей между ними, а также проектирование пользовательского интерфейса. Важно учитывать требования к производительности и масштабируемости системы, чтобы создать эффективного и удобного в использовании бота.

Таким образом, проектирование архитектуры бота требует комплексного подхода, чтобы создать эффективную и масштабируемую систему. Это включает в себя определение функциональных требований, выбор технологий, проектирование компонентов и их взаимосвязей, организацию пользовательского интерфейса, обеспечение производительности и масштабируемости, интеграцию с другими системами, аналитику и отчетность, управление и обновление.

2.3 Выбор необходимых компонентов и технологий

Для создания бота с использованием искусственного интеллекта, необходимо выбрать следующие компоненты и технологии:

1. Платформа для разработки бота: можно использовать различные платформы, такие как Microsoft Bot Framework, Google Dialogflow или AWS Lex. Они предоставляют инструменты для создания, развертывания и управления ботом[9].

2. Язык программирования: для разработки бота можно использовать языки программирования, такие как Python, JavaScript или C#. Выбор языка зависит от платформы, которую вы выбрали, и вашего опыта.

3. Компоненты обработки и понимания языка: для обработки и понимания естественного языка бота необходимо использовать NLU (понимание естественного языка) и компоненты обработки речи, такие как распознавание речи и синтез речи. Здесь можно использовать различные инструменты, такие как Natural Language Toolkit (NLTK), Google Cloud Natural Language API или Microsoft Cognitive Services.

4. Машинное обучение: для создания бота, способного обучаться на основе взаимодействия с пользователями, возможно использовать технологии машинного обучения, такие как рекуррентные нейронные сети (RNN) или глубокие нейронные сети (DNN). Для этого необходимо обучающие данные, такие как текстовые диалоги или размеченные данные.

5. Интеграция с другими системами: для того, чтобы бот выполнял различные задачи и интегрировался с другими системами, необходимо использовать API (интерфейсы программирования приложений). Например, для работы с базой данных, отправки электронной почты или выполнения действий внутри другой системы.

6. Обработка данных и аналитика: при создании бота полезно использовать инструменты для обработки данных и аналитики, чтобы анализировать данные о взаимодействии с пользователями и улучшать

функциональность бота. Здесь можно использовать инструменты, такие как Elasticsearch, Kibana или Google Analytics [5].

7. Техническая поддержка и обновления: для того, чтобы бот работал надежно и эффективно, необходимо предусмотреть техническую поддержку и обновления. Это может включать в себя мониторинг, резервное копирование данных, исправление ошибок и выпуск обновлений.

Важно отметить, что конкретные компоненты и технологии могут различаться в зависимости от требований и целей вашего бота.

Консольный интерфейс: Вы можете создать бота с консольным интерфейсом, который будет взаимодействовать с пользователем через командную строку. Это может быть полезно для прототипирования или тестирования.

Графический интерфейс пользователя (GUI): если ваш бот предназначен для взаимодействия с помощью графического интерфейса, вы можете использовать фреймворки и библиотеки для создания GUI приложений. Например, вы можете использовать PyQt, Tkinter или Electron (для создания кроссплатформенных приложений).

Интеграция с мессенджерами: если ваш бот предназначен для работы в мессенджерах, таких как Facebook Messenger, Slack или Telegram, вам потребуется использовать соответствующие API для интеграции бота с выбранным мессенджером.

Облачные платформы: для масштабируемости и доступности вашего бота вы можете использовать облачные платформы, такие как AWS, Google Cloud или Microsoft Azure. Они предоставляют инструменты для развертывания и управления вашим ботом в облаке.

Инструменты для разработки: для разработки бота вы можете использовать интегрированные среды разработки (IDE), такие как PyCharm, Visual Studio Code или IntelliJ IDEA. Они предоставляют удобную среду для написания, отладки и тестирования кода.

База данных: если ваш бот должен сохранять и управлять данными, вам

может потребоваться выбрать подходящую базу данных, такую как MongoDB, MySQL или PostgreSQL. База данных позволит хранить информацию о пользователях, истории чатов и другие важные данные.

Безопасность: Обеспечение безопасности вашего бота является важным аспектом. Вы можете использовать технологии для защиты данных и аутентификации, такие как SSL-сертификаты, OAuth или JWT (JSON Web Tokens).

Визуализация данных: если ваш бот обрабатывает большие объемы данных или предоставляет аналитику, вам потребуется инструмент для визуализации данных, такой как Matplotlib, Plotly или D3.js. Это позволит вам отображать данные в удобном и понятном виде.

Другие компоненты и технологии, которые могут быть полезны при создании бота, включают системы контроля версий (например, Git), Docker для контейнеризации приложения, системы непрерывной интеграции и развертывания (CI/CD), а также тестовые фреймворки для автоматизированного тестирования [7].

2.4 Разработка и обучение модели искусственного интеллекта

Подготовка и анализ данных для обучения модели является важным этапом машинного обучения. Этот процесс включает в себя несколько шагов, которые помогают выявить необходимые данные и обрабатывают их для более эффективного обучения модели.

Первым шагом в подготовке данных является сбор и загрузка необходимого набора данных. Набор данных, может быть, в различных форматах, таких как таблицы Excel, CSV-файлы, JSON или базы данных. Важно убедиться, что набор данных содержит все необходимые атрибуты и информацию, которую требует задача обучения модели.

В следующем шаге данные обрабатываются и очищаются от потенциальных ошибок или неполных значений. Это может включать удаление

дублирующийся записи, заполнение пустых значений или удаление выбросов. Чистые данные помогают улучшить точность и надежность модели.

Далее, проводится исследовательский анализ данных, который помогает понять особенности данных и уловить закономерности. Это включает в себя исследование статистических характеристик, визуализацию данных с помощью диаграмм и графиков, а также исследование зависимостей между атрибутами. Анализ данных позволяет выделить наиболее значимые атрибуты и определить, какие параметры могут быть наиболее полезными для обучения модели.

После этого проводится предобработка данных, включающая масштабирование, нормализацию или преобразование атрибутов для оптимизации работы модели. Это необходимо для того, чтобы разные атрибуты имели одинаковый диапазон значений и давали равный вклад в обучение модели [26].

Еще одним важным шагом является разделение данных на обучающий и тестовый наборы. Обучающий набор используется для обучения модели, в то время как тестовый набор помогает оценить качество работы модели на независимых данных. Обычно используется схема, при которой 70-80% данных используется для обучения, а оставшиеся 20-30% - для тестирования.

Наконец, производится применение специфических методов машинного обучения, таких как классификация, регрессия или кластеризация. Модели обучаются на обучающем наборе данных, а затем оцениваются на тестовом наборе. Это позволяет оценить качество и точность работы модели.

Все эти шаги работы с данными требуют тщательного подхода и внимания к деталям, чтобы получить качественную модель обучения. Правильная подготовка и анализ данных существенно влияет на результат и эффективность.

После проведения исследовательского анализа данных и предобработки, может быть необходимо выполнить дополнительные операции по подготовке данных перед обучением модели. Эти дополнительные операции зависят от конкретной задачи машинного обучения и требований модели [10].

Некоторые из возможных дополнительных операций включают кодирование категориальных переменных. Категориальные переменные представляются в форме текста или строк, например, названия стран или типы продуктов. Модели машинного обучения часто требуют числовых входных данных, поэтому необходимо преобразовать категориальные переменные в числовой формат. Это можно сделать с помощью методов, таких как кодирование с использованием фиктивных переменных или преобразование категорий в числовые значения.

Другой важной операцией является масштабирование или нормализация данных. Это может быть полезно, если различные атрибуты имеют разные единицы измерения или различный масштаб значений. Масштабирование данных помогает модели лучше справляться с такими расхождениями в данных и улучшает ее производительность.

Помимо обработки и преобразования данных, может потребоваться выполнить сокращение размерности данных. Сокращение размерности позволяет уменьшить количество признаков или атрибутов в данных, что может быть полезно для устранения шума или улучшения производительности модели. Это может быть достигнуто путем применения методов сокращения размерности, таких как главные компоненты (PCA) или методы выбора функций.

Наконец, перед обучением модели рекомендуется разделить данные на мини-партии или пакеты (mini-batch) при использовании нейронных сетей или моделей глубокого обучения. Это позволяет усреднить градиенты и обновления параметров модели, что помогает ускорить процесс обучения и повысить стабильность модели [9].

Важно отметить, что процесс подготовки и анализа данных может быть итеративным, и на каждой итерации могут быть выполнены различные операции, основанные на результате анализа или требованиях модели. Гибкость и оптимизация процесса обработки данных играют ключевую роль в достижении хороших результатов при обучении модели.

2.5 Выбор и обоснование выбора алгоритма машинного обучения

Выбор алгоритма машинного обучения зависит от конкретной задачи и имеющихся данных.

Одним из наиболее распространенных алгоритмов является метод опорных векторов (Support Vector Machine, SVM) (рисунок 7).

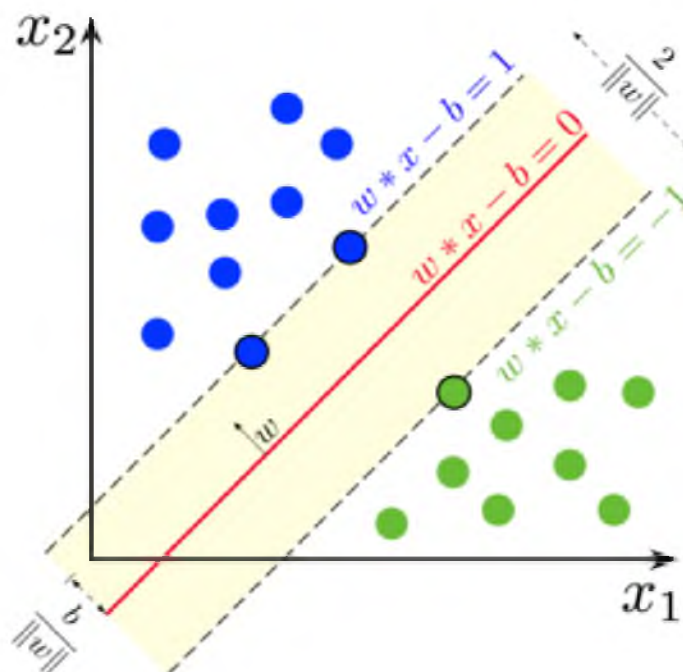


Рисунок 7– Метод опорных векторов (Support vectormachine)

Этот алгоритм отлично работает с задачами классификации и регрессии. Он основан на создании гиперплоскости, максимально разделяющей классы объектов. Метод опорных векторов хорошо обобщает данные, контролирует переобучение и обладает высокой точностью предсказания [4].

Для задачи кластеризации можно использовать алгоритм К-средних (K-means). Он разбивает набор данных на заранее заданное количество кластеров. Алгоритм итеративно обновляет центры кластеров, минимизируя сумму квадратов расстояний между объектами и их центрами. К-средних хорошо работает с данными без явно заданной структуры и позволяет легко интерпретировать результаты.

Если в данных присутствует временная зависимость, то для задачи временных рядов можно использовать алгоритм ARIMA (авторегрессионная интегрированная скользящая средняя). Он моделирует изменение значений временного ряда на основе его предыдущих значений. ARIMA способен прогнозировать будущие значения ряда и учитывать сезонность и тренды.

Если в задаче имеется большое количество признаков, то для решения можно использовать алгоритм случайного леса (RandomForest). Он основывается на комбинации нескольких решающих деревьев и позволяет выявить наиболее важные признаки. RandomForest обладает высокой скоростью обучения и предсказания, а также способностью работать с разнородными данными.

В зависимости от задачи и особенностей данных существуют и другие алгоритмы машинного обучения, такие как нейронные сети, градиентный бустинг, регрессия на основе наименьших квадратов и т.д. Основным принципом выбора алгоритма заключается в том, чтобы он наиболее эффективно решал поставленную задачу и работал с имеющимися данными.

Если задача связана с обработкой текстовых данных, то можно воспользоваться алгоритмом машинного обучения на основе метода bagofwords (мешок слов). Этот метод предполагает преобразование текстовых фрагментов в числовые векторы, считая количество вхождений каждого слова в текст. Для классификации текстовых данных можно использовать алгоритм наивного байесовского классификатора, который основан на принципе байесовской вероятности и хорошо справляется с задачами категоризации текстов.

В случае, когда данные обладают временной или пространственной зависимостью, можно использовать рекуррентные нейронные сети (RNN) или сверточные нейронные сети (CNN). RNN позволяют учесть контекст и последовательность данных, а CNN эффективно обрабатывают данные в пространственной сетке, например изображения.

Если задача состоит в обнаружении аномалий в данных, то подходящим алгоритмом будет одно классовая машина опорных векторов (One-Class SVM).

Этот алгоритм обучается представлять только один класс данных и ищет объекты, которые не соответствуют этому классу, то есть являются аномалиями.

Для решения задачи ранжирования можно использовать градиентный бустинг (Gradient Boosting). Этот алгоритм работает с набором базовых моделей, объединенных в ансамбль, и последовательно добавляет модели, которые исправляют ошибки предыдущих моделей. Градиентный бустинг хорошо справляется с ранжированием объектов по их значимости или вероятности.

В общем случае, выбор алгоритма машинного обучения зависит от особенностей данных, размера выборки, степени сложности задачи, требуемой точности предсказания и ресурсов, которые можно выделить для обучения и применения модели. Важно провести исследование и эксперименты с несколькими алгоритмами для выбора, наиболее подходящего в данной ситуации [10].

2.6 Разработка и обучение модели на выбранных данных

Разработка и обучение бота с использованием искусственного интеллекта модели на выбранных данных также включает несколько этапов:

1. Определение целей и задачи бота: нужно определить, какие задачи бот будет выполнять и какие данные для этого требуются. Например, задачей может быть ответ на вопросы пользователей или выполнение каких-то задач по запросу.

2. Сбор и подготовка данных: необходимо собрать данные, которые потребуются для обучения и функционирования бота. Это могут быть текстовые данные, диалоговые сценарии, базы знаний или любая другая информация, которая поможет боту выполнить свою задачу. Необходимо также произвести предобработку данных, включающую удаление шума, нормализацию текста и другие необходимые преобразования.

3. Выбор модели и архитектуры бота: на основе задачи и данных нужно выбрать модель машинного обучения, которая лучше всего соответствует требованиям. Например, для задачи вопросно-ответной системы можно использовать методы обработки естественного языка (Natural Language Processing, NLP), такие как рекуррентные нейронные сети (RNN) или трансформеры (transformers).

4. Обучение модели: после выбора модели происходит обучение бота на подготовленных данных. Это может быть выполнено с использованием методов градиентного спуска, оптимизации или других подходов. Во время обучения модели можно провести мониторинг процесса и анализировать результаты.

5. Оценка и настройка модели: после обучения модели ее необходимо оценить, чтобы определить ее производительность и качество. Можно использовать метрики, такие как точность, переплески, BLEU score и т.д. Если результаты неудовлетворительные, можно произвести настройку модели, изменяя ее гиперпараметры и архитектуру.

6. Тестирование и внедрение: после оценки и настройки модели необходимо протестировать бота на независимых данных и оценить его способность выполнять поставленные задачи. Если бот успешно проходит тестирование, то его можно внедрить для использования.

7. Обновление и поддержка: разработка бота — это постоянный процесс. После внедрения бота необходимо оценивать его работу в реальной среде, анализировать обратную связь от пользователей и вносить необходимые изменения и дополнения в модель или данные. Это поможет улучшить качество работы бота и его способность адаптироваться к меняющимся условиям.

Разработка и обучение бота с использованием модели искусственного интеллекта может быть сложным и требовательным процессом, но с правильным подходом и компетентной командой может привести к созданию мощного и полезного инструмента.

8. Интеграция бота: после успешного тестирования и внедрения необходимо интегрировать бота в выбранную платформу или приложение. Это может включать создание API для взаимодействия с ботом, настройку пользовательского интерфейса или интеграцию с другими системами.

9. Обратная связь и улучшение: после внедрения и использования бота в реальных условиях важно собирать обратную связь от пользователей. Это поможет определить слабые места и улучшить модель. Можно использовать методы анализа данных, такие как просмотр и классификация пользовательских отзывов, для получения ценной информации о производительности и качестве работы бота.

10. Мониторинг и обновление: важно мониторить работу бота после его внедрения и обновлять его при необходимости. Это может включать обновление данных, переобучение модели или внесение изменений в алгоритмы работы бота. Также важно регулярно проверять работоспособность и эффективность бота и вносить корректировки в его алгоритмы при необходимости [7, 18].

3. Реализация и интеграция бота

3.1 Разработка бота с использованием выбранных технологий

Для создания бота с использованием искусственного интеллекта был выбран конструктор Just AI Aimylogic, который предоставляет удобные средства для создания, настройки и интеграции ботов [30]. Существует несколько шагов, которые помогут в реализации и интеграции бота через этот конструктор:

1. Регистрация в Aimylogic и создание нового проекта для данного бота. Будет предложено выбрать платный или бесплатный тарифный план в зависимости от конечных требований.

2. Создание сценария для бота, используя графический интерфейс Aimylogic. Необходимо определить различные намерения (intentions), сущности (entities), ответы бота и логику переходов между ними.

3. Настройка интеграции с различными каналами общения, такими как сайт, мессенджеры (например, Facebook Messenger, Telegram) или телефония. Aimylogic предоставляет готовые шаблоны для различных платформ, что упрощает процесс настройки.

4. Подключение к искусственному интеллекту Just AI (платформе, на которой работает Aimylogic). В ней можно обучить бота распознавать и понимать пользовательские запросы с помощью обученных моделей нейронных сетей или установить предопределенное поведение с помощью правил.

5. Тестировка бота и его интеграцию с выбранными каналами общения. Можно использовать эмуляторы Aimylogic для этой цели или проверять бота непосредственно в каналах связи.

6. После успешного тестирования необходимо опубликовать бота и запустить его в работу.

В результате выйдет готовый бот, который будет общаться с пользователями через выбранные каналы общения. Далее

можно продолжить настраивать бота, добавлять новые функции и улучшать его с помощью Aimylogic [21].

Для этого перейдём на их сайт и попробуем создать своего бота:

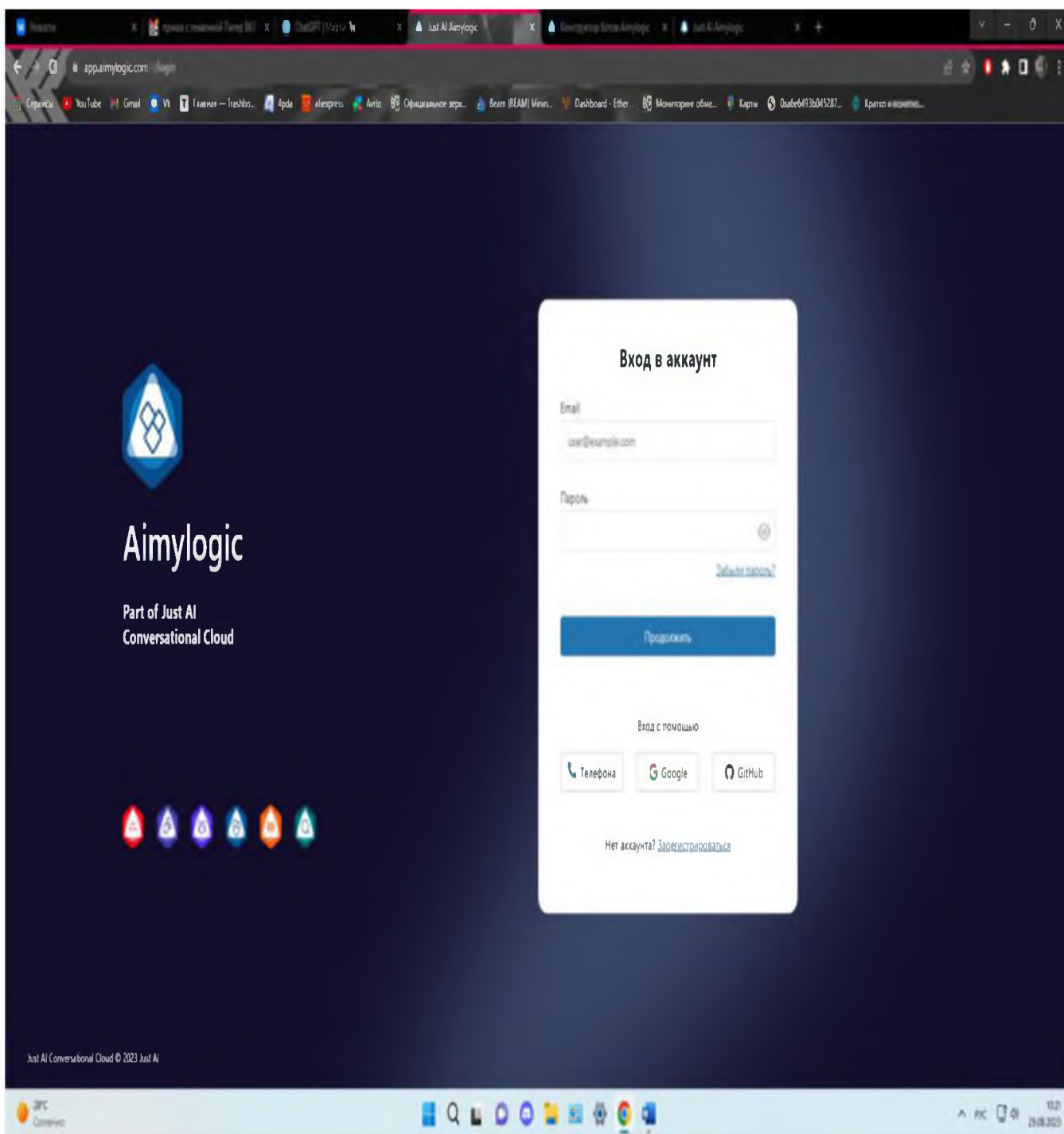


Рисунок 8 – Сайт Aimylogic

Для разработки бота нам следует зарегистрироваться на сайте Aimylogic (рисунок 8). Далее выбираем создания бота в редакторе навыков показанный на рисунке 9.

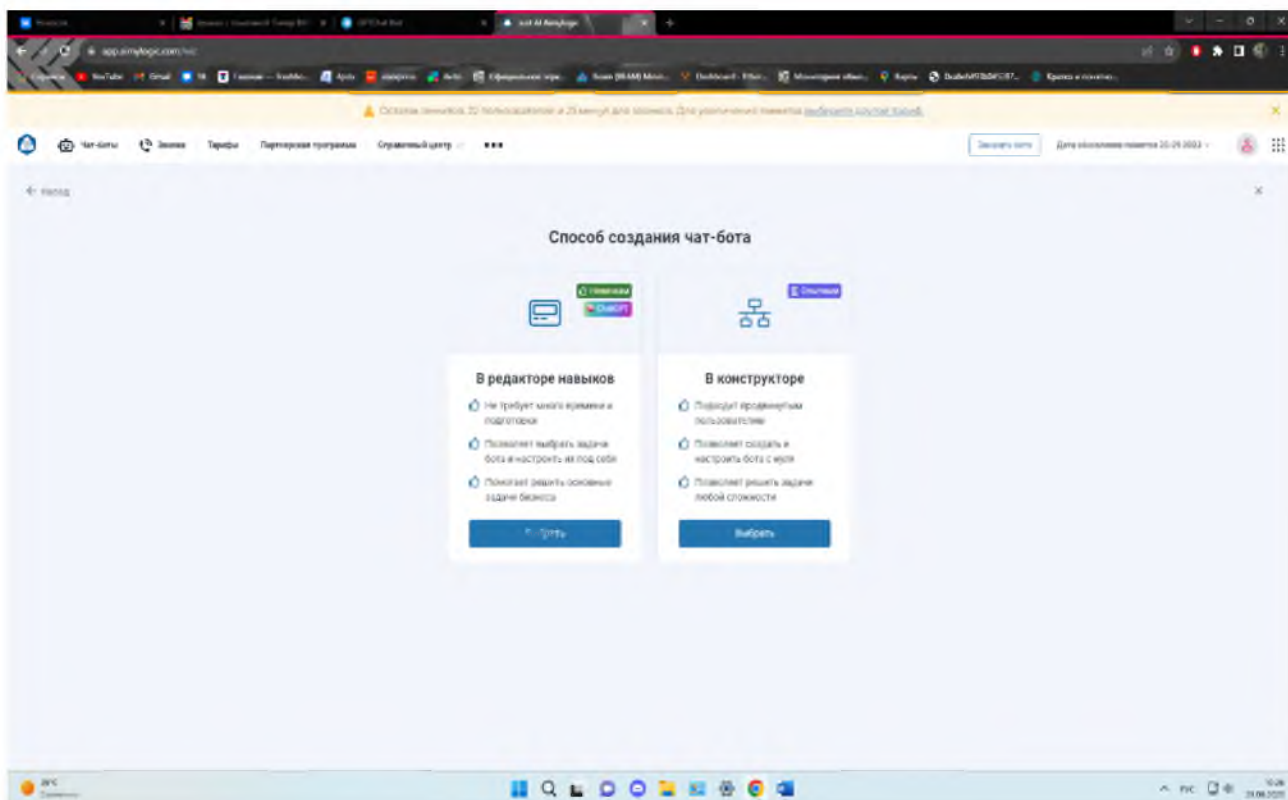


Рисунок 9 – Способы создания чат-бота

Aimylogic предоставляет несколько способов создания чат-бота на вашем личном сайте. Вот некоторые из них:

1. Создание скрипта с нуля: можете создать скрипт для своего чат-бота с помощью графического интерфейса Aimylogic. Здесь вы можете определить намерения, сущности и логику диалога вашего бота, используя готовые блоки или создавая их самостоятельно. Затем вы можете настроить веб-интеграцию для встраивания вашего бота на ваш сайт.

2. Импорт существующих скриптов: если у вас уже есть готовый скрипт для бота, вы можете импортировать его в Aimylogic и настроить веб-интеграцию для размещения на вашем сайте. Aimylogic поддерживает импорт скриптов из других платформ чат-ботов, таких как Dialogflow, Watson, Яндекс. Диалоги и т. д.

3. Использование готовых шаблонов: Aimylogic предоставляет готовые шаблоны для создания чат-ботов различного типа. Там можете выбрать подходящий шаблон и настроить его под ваши потребности. Это упростит

процесс создания бота и ускорит внедрение на сайте.

4. Использование пред обученных базовых моделей: Aimylogic предоставляет базовые модели ботов, которые уже обучены на различные намерения и сущности. Вы можете использовать эти модели, чтобы быстро создать и настроить простых чат-ботов без необходимости обучения с нуля.

После создания и настройки бота, можете воспользоваться инструментами Aimylogic для интеграции сайте и с использовать готовые веб-виджеты или API Aimylogic, чтобы встраивать и взаимодействовать ботом.

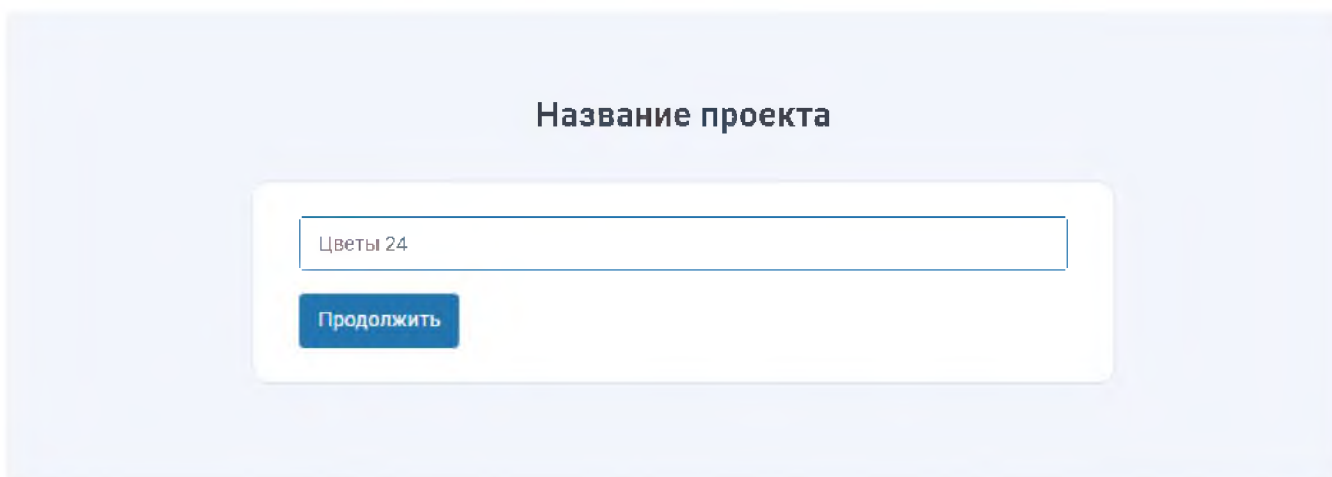


Рисунок 10 -Название проекта

Придумываем название для нашего бота (рисунок 10).

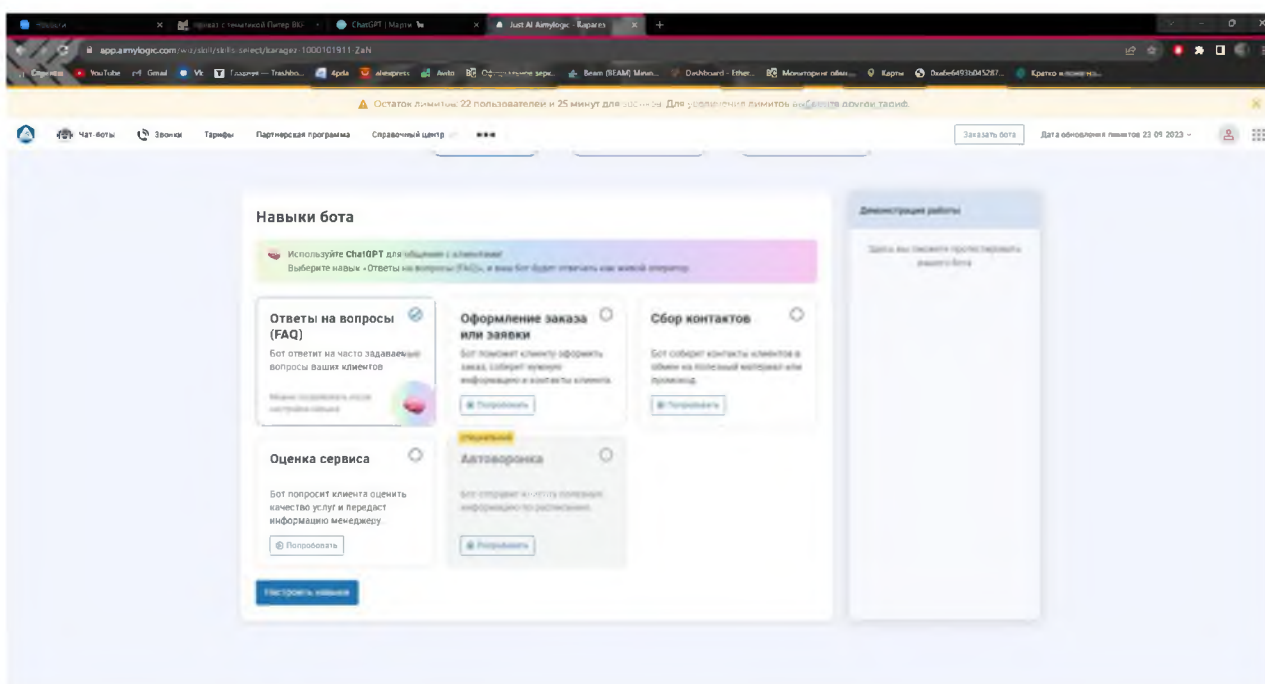


Рисунок 11 – Навыки бота.

Для нашей цели выбираем «ответы на вопросы» (рисунок 11)

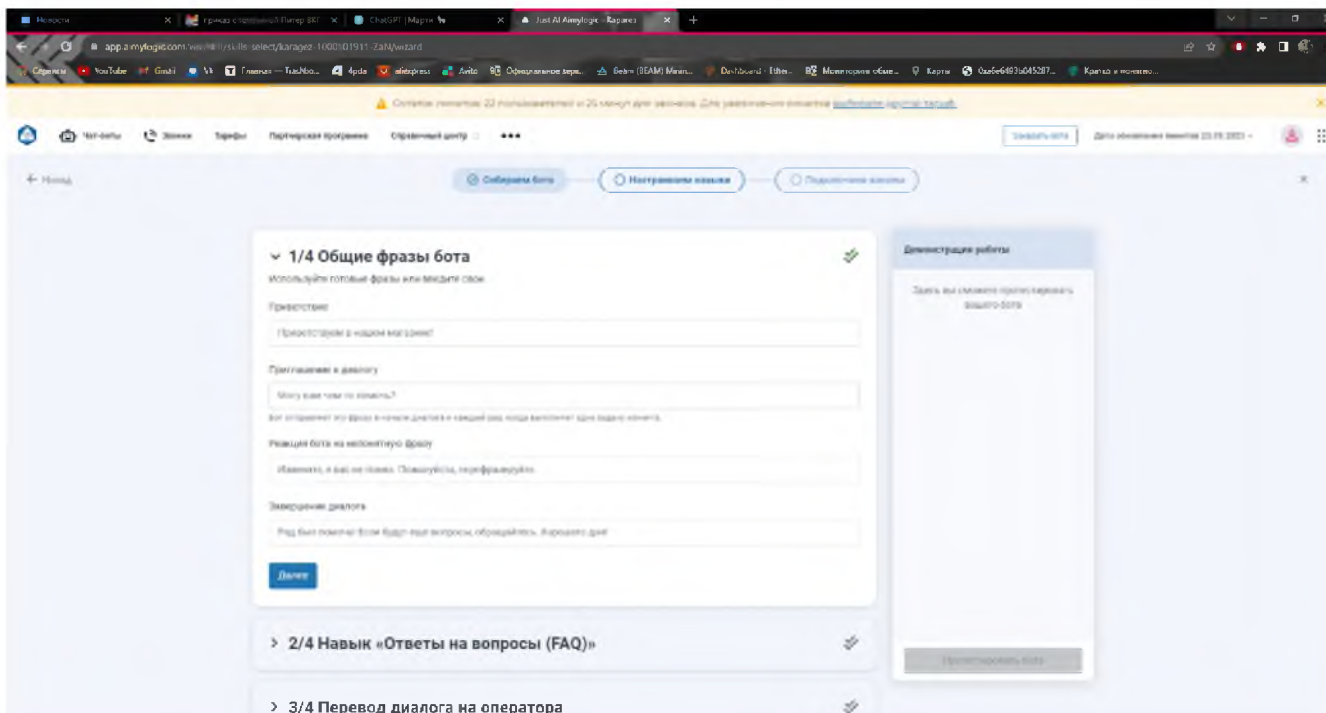


Рисунок 12 – Создание бота

В настройке навыков мы можем создавать общие фразы, приветствия, приглашению к диалогу, реакцию бота на непонятную фразу и завершения диалога (рисунок 12).

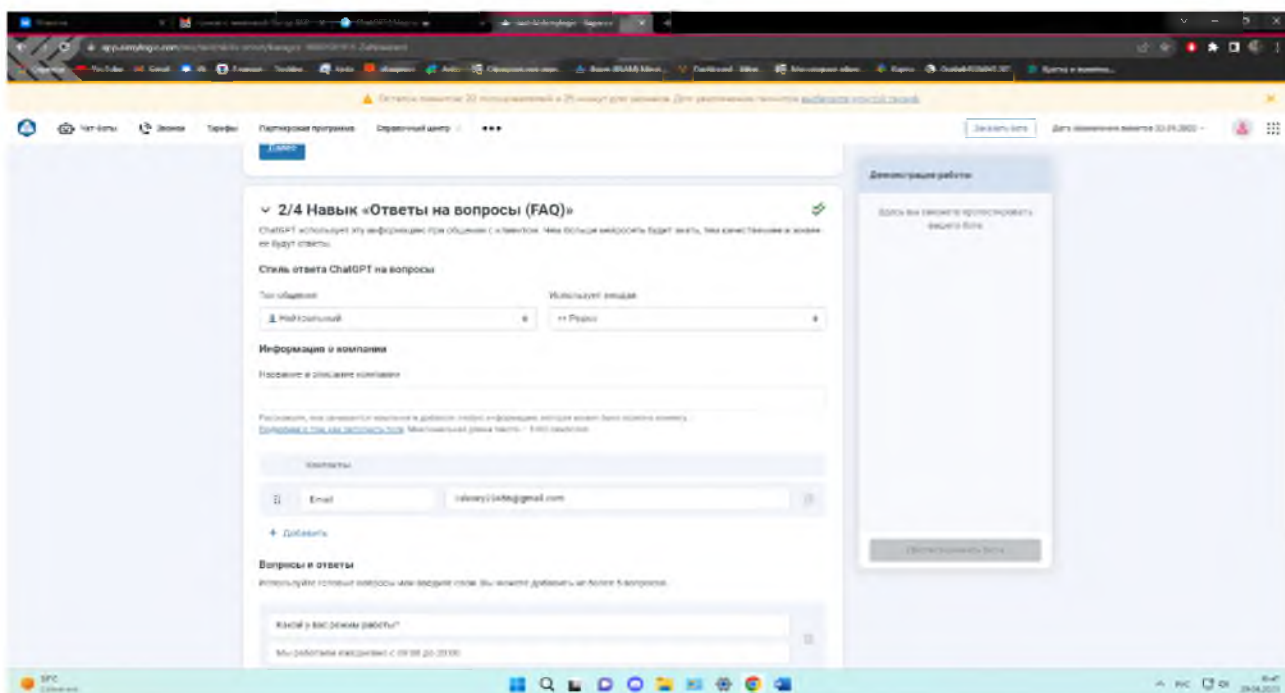


Рисунок 13 – Навык бота

В навыке бота «Ответы на вопросы» мы можем создать ему стиль общения, информацию о боте, контакты для связи, а также выбрать режим работы бота (рисунок 13).

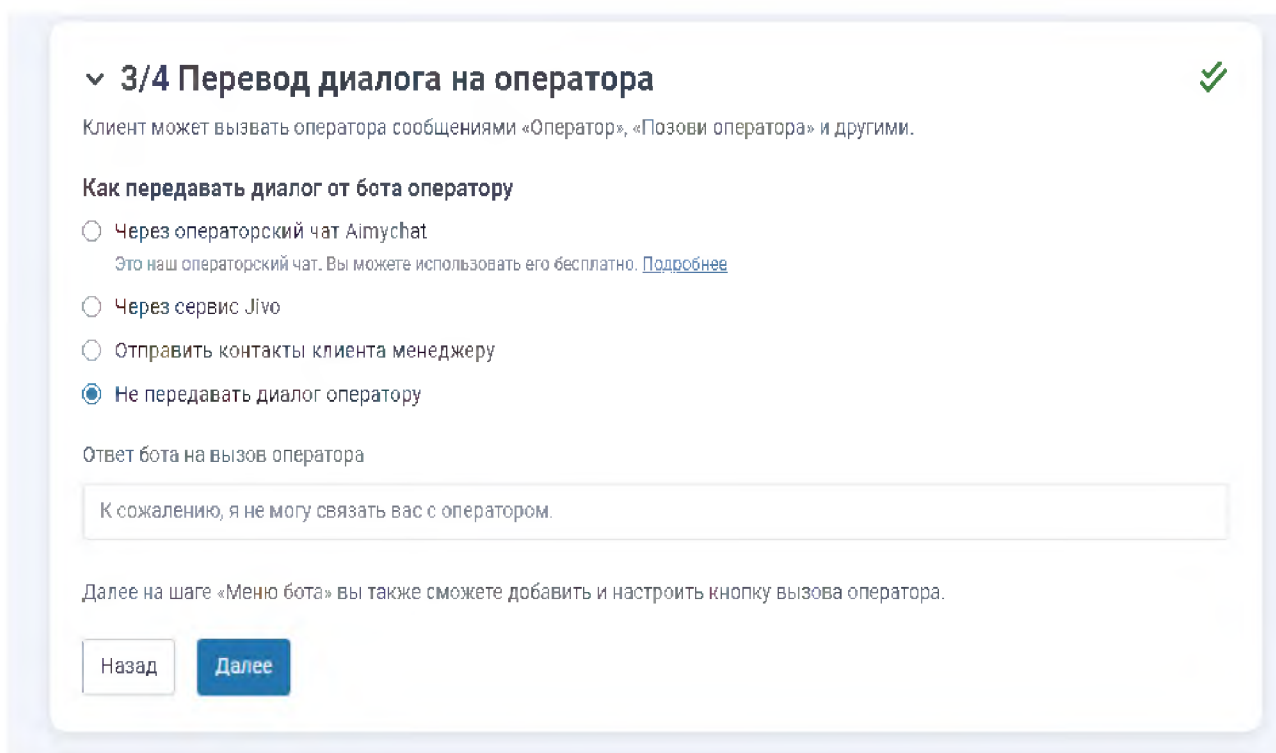


Рисунок 14 – Перевод диалога на оператора

Клиент может вызвать оператора сообщениями «Оператор», «позови оператора» и другими (рисунок 14).

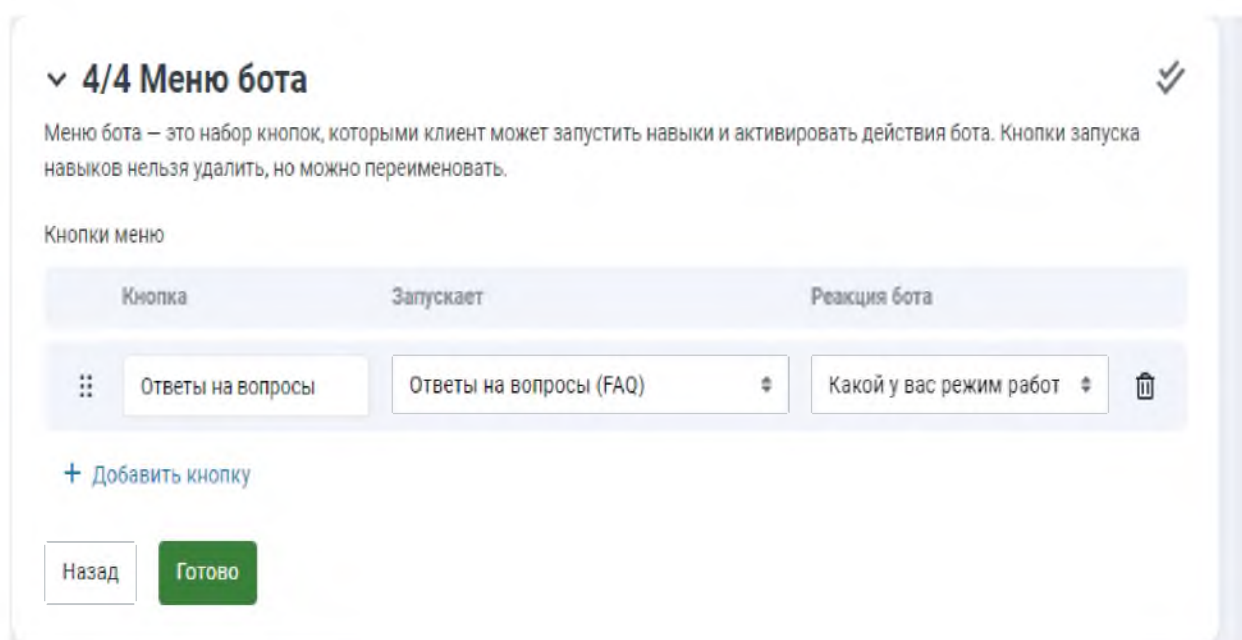


Рисунок 15 – Меню бота

Меню бота — это набор кнопок, которыми клиент может запустить навыки и активировать действия бота. Кнопки запуска навыков нельзя удалить, но можно переименовать (рисунок 15)

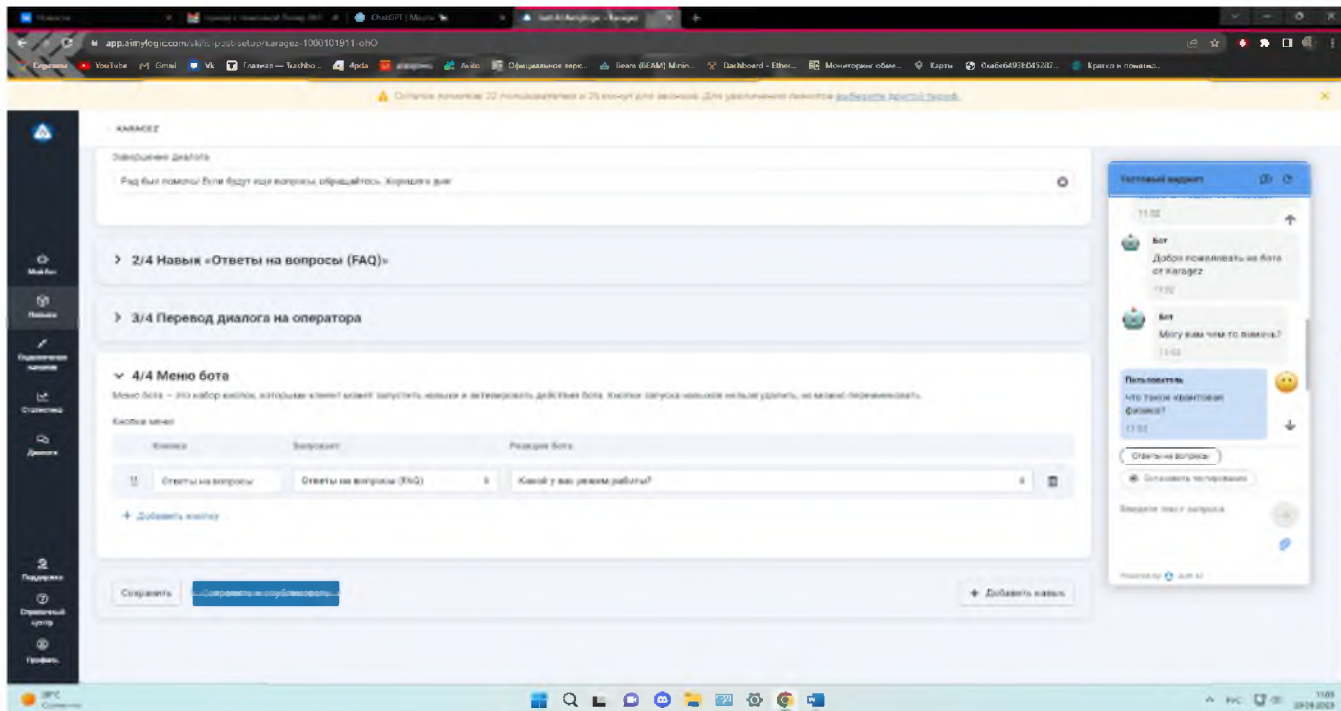


Рисунок 16 – Тест созданного бота

В тестовом виджете мы можем наблюдать приветствие бота и задавать ему вопросы, на которые он спокойно отвечает (рисунок 16)

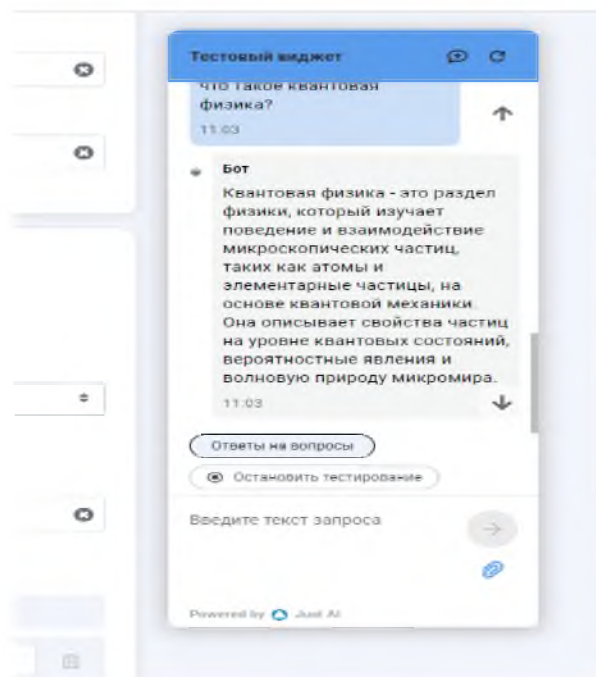


Рисунок 17 – Проверка на вопрос бота.

Проверив бота (рисунок 17) и убедившись в его работоспособности, могу преступать к следующей задаче.

3.2 Интеграция бота в выбранную платформу или систему.

В системе Aimylogic, можно выбрать следующие подключения с клиентами:

1. На самом сайте.
2. Telegram
3. ВКонтакте
4. WhatsApp
5. Viber

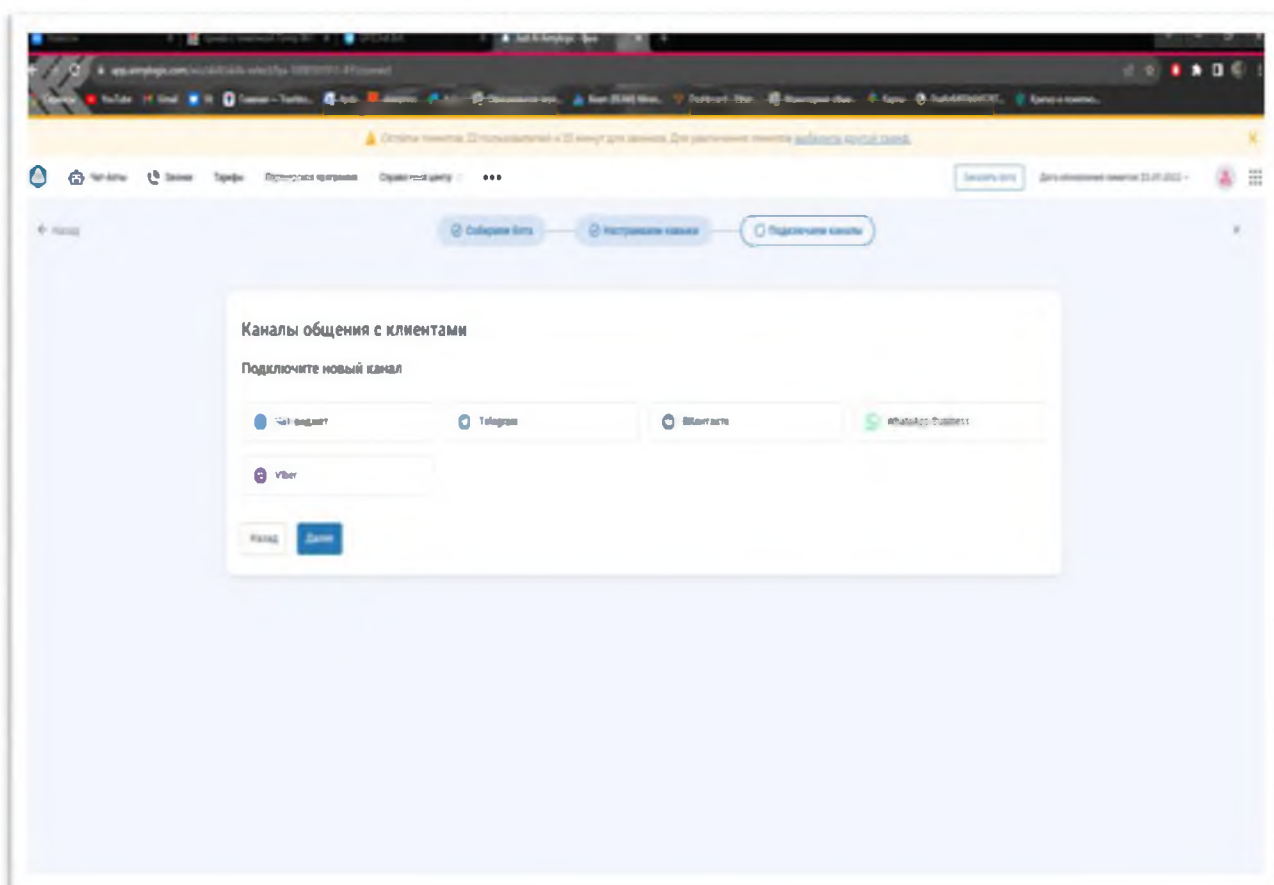


Рисунок 18 – Каналы общения

Среди предложенного списка, самым популярным и удобным является Telegram, его мы и будем использовать.

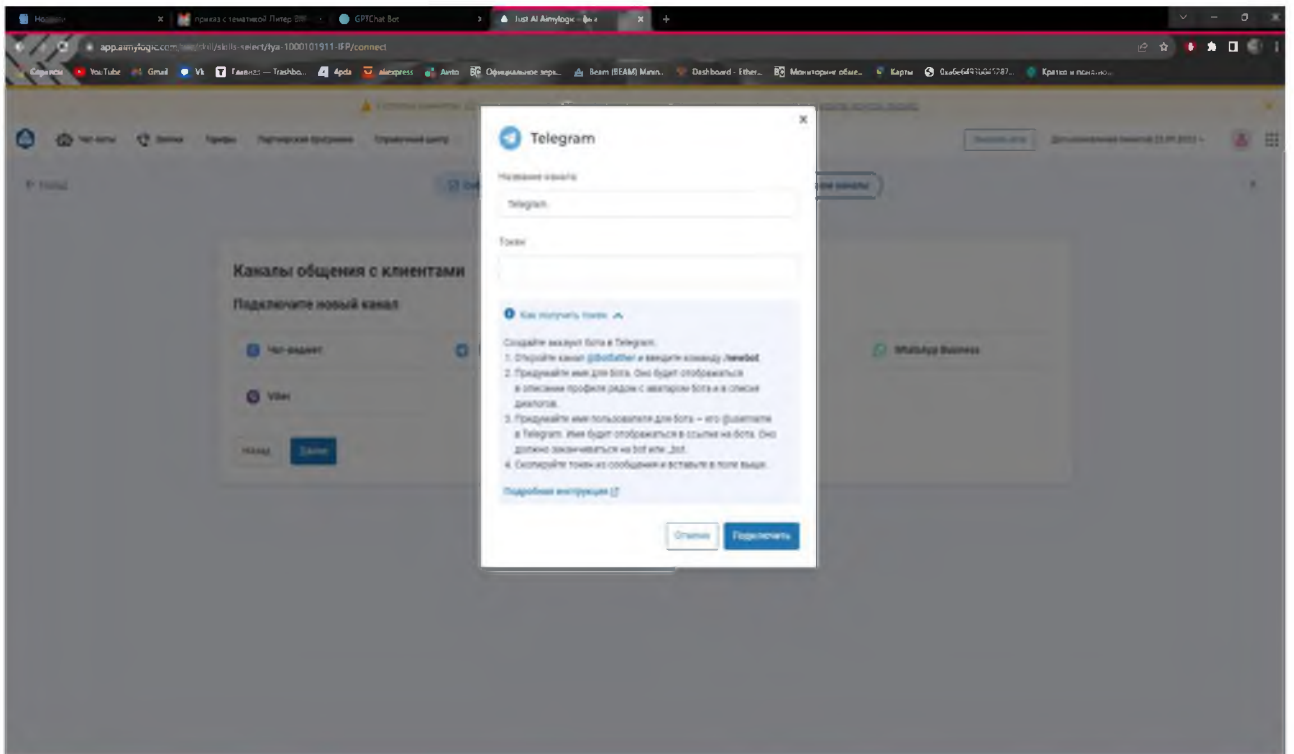


Рисунок 19 – Создание токена

Чтобы получить токен, нам надо:

1. Перейти на платформу Телеграм (рисунок 19)
2. Найти бота – BotFather
3. Написать ему –NewBot
4. Следовать его инструкции (Название бота и т.д.)

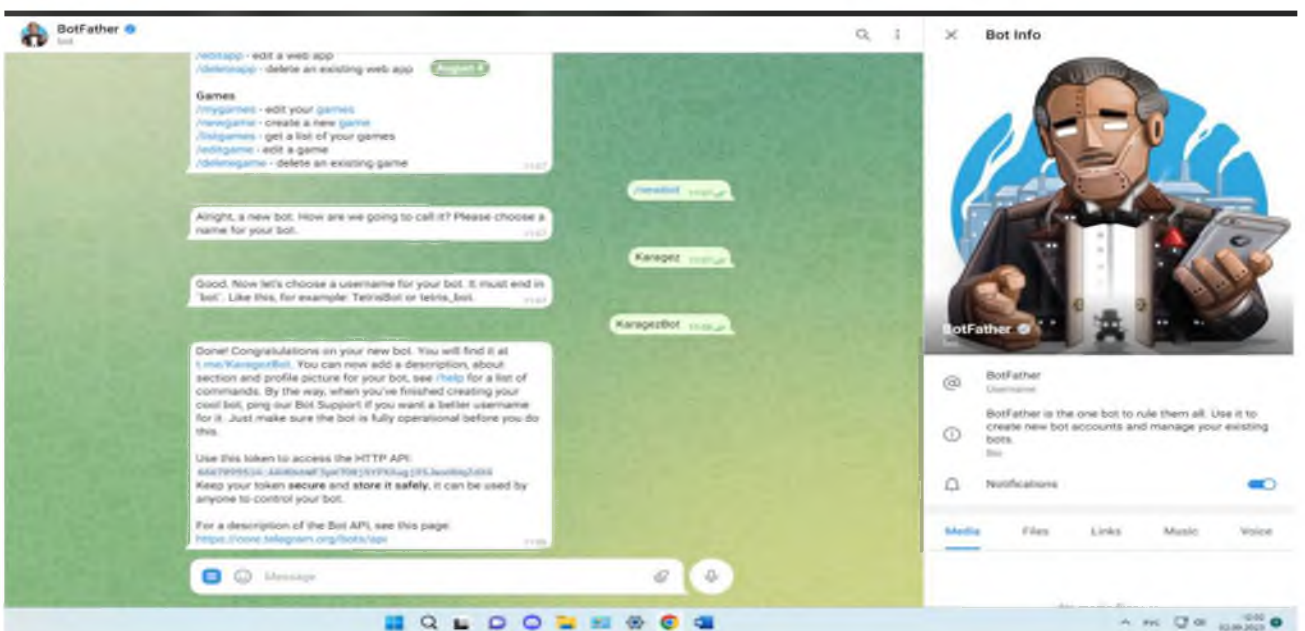


Рисунок 20 – Создание бота

После создание бота в BotFather, у нас получится пустой бот. Чтобы активировать бота нам нужно полученный токен от бота BotFather, вставить на сайте Aimylogic (рисунок 19)

3.3 Тестирование бота

Созданный бот, может помогать вам не только в коде, но и отвечать быстро на вопросы. Главным преимуществом - не надо будет по всему интернету искать вопрос, ведь бот сделает всё за вас, отталкиваясь от запроса (рисунок 21)

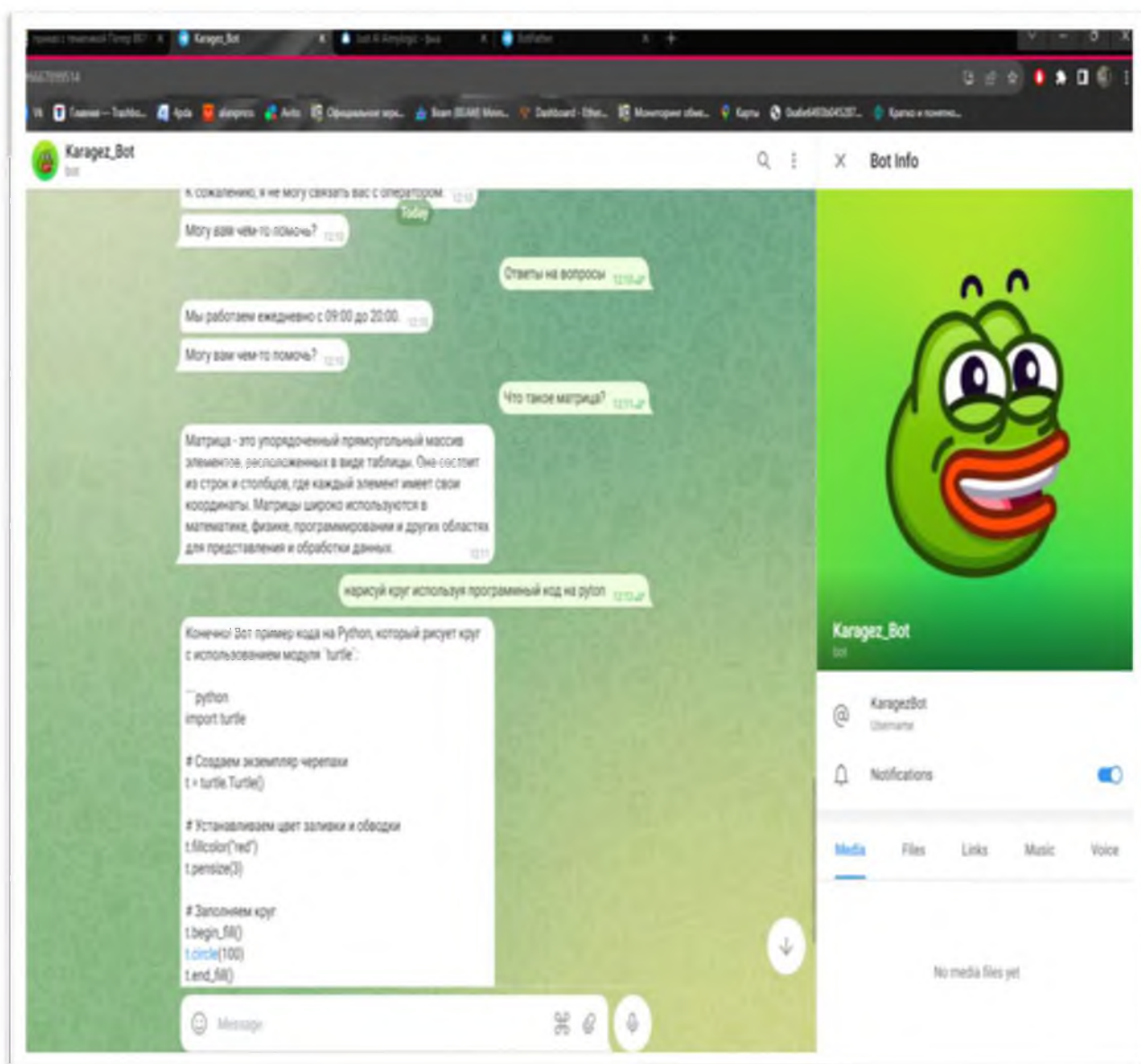


Рисунок 21 – Проверка бота в действии

Проверить данные можно рассмотрев рисунок 21

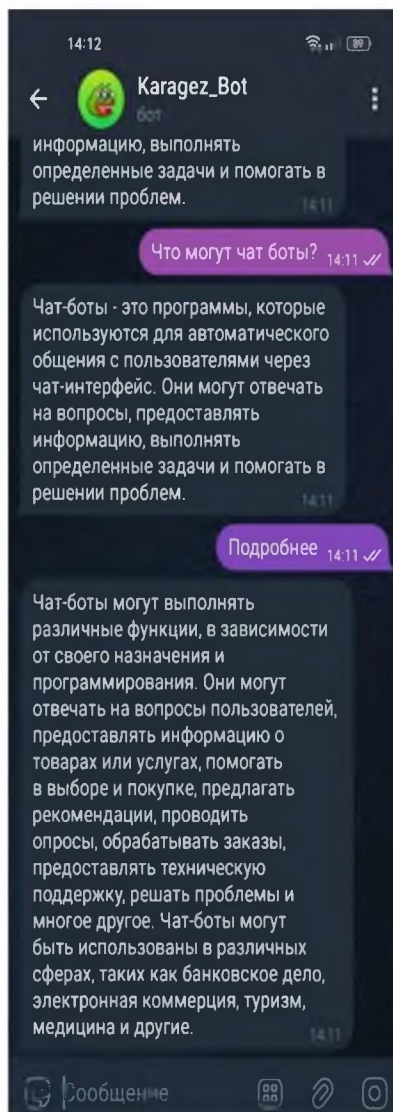


Рисунок 22 – Проверка бота на смартфоне.

Данный бот является аналогом бота ChatGPT и использует немного старую версию алгоритма [22].

В ходе тестирования рекомендуется проверить следующие аспекты:

1. Корректную обработку входных данных и взаимодействие с пользователем.
2. Правильность работы логики бота, включая условия и переходы между блоками диалога.
3. Качество и полноту ответов бота на вопросы пользователей.

После завершения тестирования обязательно проверьте логи бота в Aimylogic для выявления любых проблем или ошибок (рисунок 23)

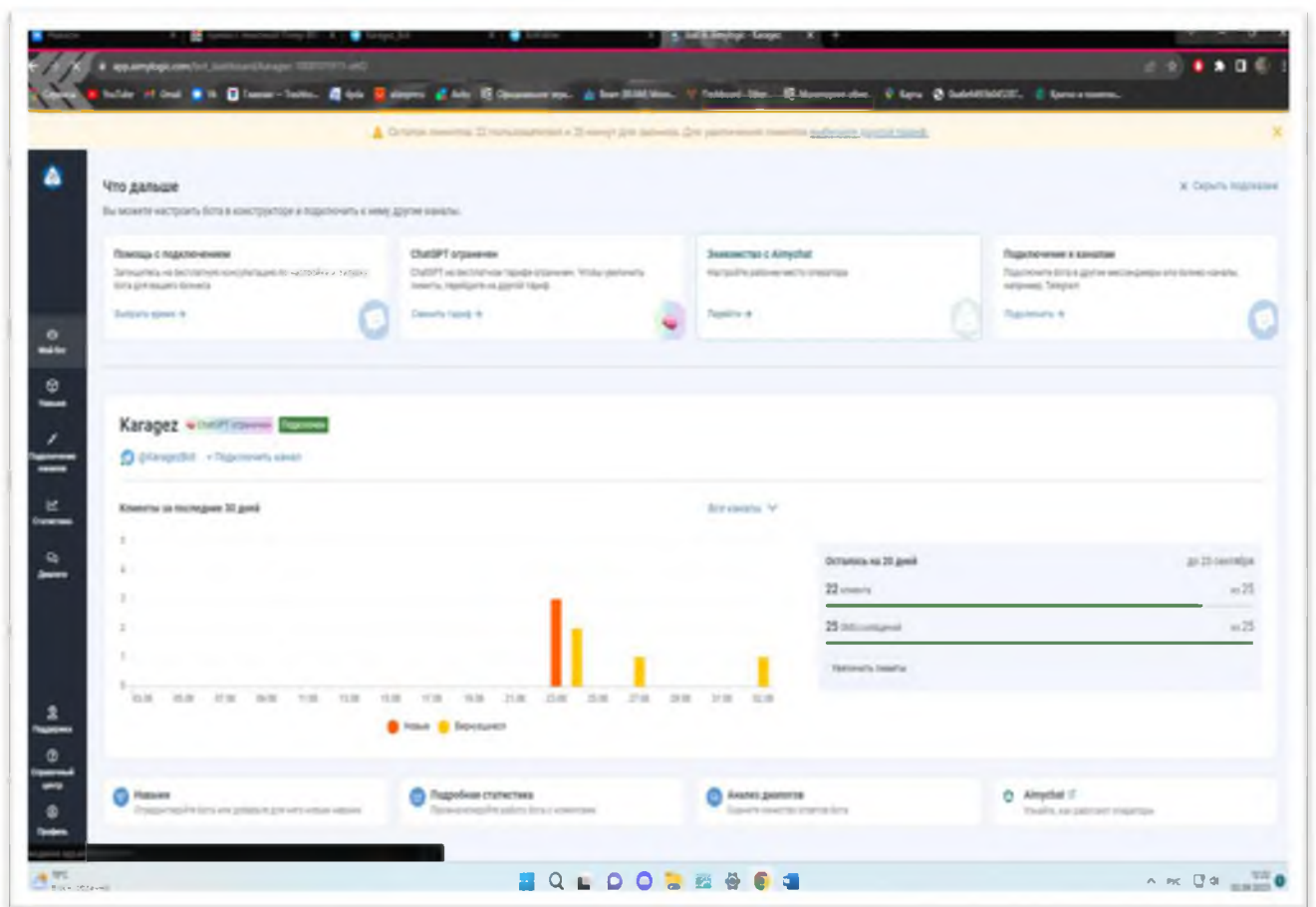


Рисунок 23 – Информация о боте

На сайте AImylogics можно следить за стабильностью и работе бота, в любой момент можно изменять настройки бота и т.д. (рисунок 23)

Заключение

В заключение, создание бота с использованием искусственного интеллекта является важным и перспективным направлением развития технологий. Благодаря применению искусственного интеллекта, боты становятся все более интеллектуальными и способными адаптироваться к потребностям пользователей.

Создание бота с использованием искусственного интеллекта требует определенных навыков и знаний, включая знание языков программирования, алгоритмов машинного обучения и обработки естественного языка. Однако, благодаря развитию инструментов и платформ для разработки ботов, этот процесс становится все более доступным для широкой аудитории.

Боты с искусственным интеллектом имеют широкий спектр применений, от обслуживания клиентов в сфере бизнеса до автоматизации рутинных задач. Они способны эффективно отвечать на вопросы пользователей, предоставлять рекомендации, анализировать данные и многое другое.

Однако, несмотря на все преимущества, создание бота с искусственным интеллектом также включает в себя некоторые риски и ограничения. Например, боты могут оказаться неправильно обученными или уязвимыми к атакам, что может привести к некачественному обслуживанию клиентов или утечке конфиденциальной информации.

В целом, создание бота с использованием искусственного интеллекта представляет большие возможности для улучшения пользовательского опыта и автоматизации задач. Однако, необходимо придерживаться строгих стандартов безопасности и этического использования, чтобы обеспечить надежность и доверие пользователей к таким ботам.

Список литературы

1. Amazon Lex - сервис искусственного интеллекта, предоставляемый Amazon Web Services (AWS), для разработки голосовых и текстовых чат-ботов.[Электронный ресурс]. <https://aws.amazon.com/lex/> (дата обращения: 29.11.2023)
2. Chatfuel - платформа для создания чат-ботов в Facebook Messenger с использованием блочного интерфейса и функций машинного обучения.[Электронный ресурс]. <https://chatfuel.com/> (дата обращения: 10.11.2023)
3. Dialogflow - платформа Google Cloud для разработки чат-ботов и голосовых приложений с помощью машинного обучения.[Электронный ресурс]. <https://dialogflow.cloud.google.com/> (дата обращения: 11.10.2023)
4. IBM Watson - платформа ИИ и облачных сервисов, предоставляющая мощные инструменты для разработки ботов.[Электронный ресурс]. <https://www.ibm.com/watson>. (дата обращения: 10.11.2023)
5. Microsoft BotBuilder - набор инструментов и библиотек с открытым исходным кодом от Microsoft для создания интеллектуальных ботов.[Электронный ресурс]. <https://github.com/microsoft/botbuilder> (дата обращения: 11.09.2023)
6. Microsoft Bot Framework - фреймворк для создания ботов с использованием искусственного интеллекта на платформе Microsoft Azure.[Электронный ресурс]. <https://dev.botframework.com/>(дата обращения: 11.09.2023)
7. PyTorch - фреймворк для глубокого обучения с открытым исходным кодом, который предоставляет инструменты для создания и обучения нейронных сетей.[Электронный ресурс]. <https://pytorch.org/>(дата обращения: 11.11.2023)
8. Rasa - открытая платформа для создания разговорных агентов с поддержкой искусственного интеллекта.[Электронный ресурс]. <https://rasa.com/>(дата обращения: 11.12.2023)

9. TensorFlow - библиотека с открытым исходным кодом для разработки и обучения искусственных нейронных сетей. [Электронный ресурс]. <https://www.tensorflow.org/> (дата обращения: 10.10.2023)

10. Wit.ai - платформа для разработки натурального языка и создания интеллектуальных ассистентов. [Электронный ресурс]. <https://wit.ai/> (дата обращения: 10.11.2023)

11. Oracle Digital Assistant - платформа искусственного интеллекта, предоставляемая компанией Oracle, для разработки голосовых и текстовых чат-ботов. [Электронный ресурс]. <https://www.oracle.com/ru/chatbots/digital-assistant/> (дата обращения: 29.11.2023)

12. Google Dialogflow - платформа искусственного интеллекта, предоставляемая компанией Google, для создания голосовых и текстовых чат-ботов. [Электронный ресурс]. <https://cloud.google.com/dialogflow> (дата обращения: 29.11.2023)

13. SAP Conversational AI - платформа искусственного интеллекта, предоставляемая компанией SAP, для разработки голосовых и текстовых чат-ботов. [Электронный ресурс]. <https://cai.tools.sap/> (дата обращения: 29.11.2023)

14. Nuance Communications - компания, специализирующаяся на разработке решений искусственного интеллекта для голосовых и текстовых чат-ботов. [Электронный ресурс]. <https://www.nuance.com/> (дата обращения: 29.11.2023)

15. Нейросеть Chat GPT на русском –тестирование онлайн текстовых чат-ботов. [Электронный ресурс]. <https://gpt-chatbot.ru/> (дата обращения: 29.11.2023)

16. Botpress - открытая платформа для разработки и развертывания чат-ботов с использованием искусственного интеллекта. [Электронный ресурс]. <https://botpress.com/> (дата обращения: 11.12.2023)

17. SnatchBot - платформа для создания чат-ботов с помощью блочного интерфейса и интеграции с различными мессенджерами. [Электронный ресурс]. <https://snatchbot.me/> (дата обращения: 11.12.2023)

18. Pandorabots - платформа для создания и развертывания разговорных

агентов с использованием искусственного интеллекта. [Электронный ресурс]. <https://www.pandorabots.com/> (дата обращения: 11.12.2023)

19. OpenAI - исследовательская организация, разрабатывающая инструменты и модели для искусственного интеллекта, включая чат-ботов. [Электронный ресурс]. <https://openai.com/> (дата обращения: 11.12.2023)

20. Botsify - платформа для разработки чат-ботов с использованием блочного интерфейса и возможностей искусственного интеллекта. [Электронный ресурс]. <https://botsify.com/> (дата обращения: 11.12.2023)

21. ManyChat - платформа для создания чат-ботов в Facebook Messenger с помощью блочного интерфейса и функций автоматизации. [Электронный ресурс]. <https://manychat.com/> (дата обращения: 11.12.2023)

22. ChatBot - платформа для создания чат-ботов с использованием искусственного интеллекта и функций автоматизации. [Электронный ресурс]. <https://www.chatbot.com/> (дата обращения: 11.12.2023)

23. Microsoft Cognitive Services - набор API и инструментов от Microsoft для разработки интеллектуальных приложений, включая чат-ботов. [Электронный ресурс]. <https://azure.microsoft.com/services/cognitive-services/> (дата обращения: 11.12.2023)

24. Salesforce Einstein Bots - интегрированный инструмент Salesforce для создания и развертывания чат-ботов с функциями искусственного интеллекта. [Электронный ресурс]. <https://www.salesforce.com/products/einstein/overview/> (дата обращения: 11.12.2023)

25. BotXO - платформа для создания и развертывания чат-ботов с функциями искусственного интеллекта. [Электронный ресурс]. <https://www.botxo.co/> (дата обращения: 11.12.2023)

26. Anthome.ru - анализ новых функций Plugins и Web browsing. [Электронный ресурс]. <https://www.anthome.ru/> (дата обращения: 11.12.2023)

27. SWOT анализ - Материал из Википедии [Электронный ресурс]. <https://ru.wikipedia.org/wiki/SWOT/> (дата обращения: 11.12.2023)

28. Loginom - Классификация данных при помощи нейронных сетей

[Электронный ресурс]. <https://loginom.ru/blog/neural-classification/>(дата обращения: 11.12.2023)

29. Платформа aimylogic – Конструктор ботов[Электронный ресурс]. <https://aimylogic.com/>(дата обращения: 12.12.2023)

30. Knowledge - База знаний по конструктору Aimylogic[Электронный ресурс]. <https://knowledge.just-ai.com/docs/ru/products/aimylogic/> (дата обращения: 12.12.2023)