

Муниципальное общеобразовательное учреждение

«Гимназия №3»



**Региональная конференция по научно-техническому творчеству
школьников «Лабиринты науки»**

Секция: Информационно-телекоммуникационные технологии

Исследовательская работа на тему:

«Оптимизация процесса поиска и сбора информации путем создания чат-ботов: на примере создания чат-бота для Детского технопарка «Кванториум» на базе МОУ «Гимназия №3»

Выполнили: обучающиеся 10 «А» класса

(технологический профиль)

Новиков Илья Олегович,

Девятаев Константин Васильевич.

Научный руководитель: Арсеньева Диана Павловна,

педагог дополнительного образования

Детского технопарка «Кванториум» на базе МОУ «Гимназия №3»

Ярославль, 2022

Оглавление

Введение	3	
• Анализ способов оптимизации	5	
• Выбор и описание базового варианта	7	
• Технология создания Telegram-бота	8	
Заключение	9	
Словарь терминов	10	
Список литературы	10	
Приложения	12	

Введение

В настоящее время интернет является неотъемлемой частью жизни практически каждого человека. Он дал людям возможность гораздо быстрее и удобнее обмениваться информацией. Процесс развития интернет-технологий является толчком для поиска инновационных решений, с помощью которых появляется возможность повышения удобства и эффективности в том числе и образовательного процесса. Это обусловлено возможностью внедрения технологичных решений, настроенных на автоматизацию некоторых процессов, связанных с поиском и обработкой информации, как со стороны сотрудников учебного заведения, так и со стороны обучающихся.

Важным принципом в современной системе образования является информационная открытость и доступность. Образовательные учреждения используют интернет-пространство в целях обеспечения коммуникации между всеми участниками образовательного процесса.

У обучающихся Детского технопарка «Кванториум» на базе МОУ «Гимназия №3» (далее Школьный Кванториум), который открылся совсем недавно - 8 сентября 2022 года, появилась следующая проблема: необходимость удобно и быстро просматривать информацию о Кванториуме, расписании занятий, объявления и новости, что на данный момент реализовано только частично. С 2.08 по 30.08.2022 г. нами был проведен опрос, в котором приняли участие 302 гимназиста. По результатам опроса, 45,9% обучающихся считают самым удобным способом получения информации о Кванториуме – мессенджер Telegram.

Актуальность темы проектной работы заключается в необходимости использования онлайн-сервисов для взаимодействия между образовательной организацией и обучающимися/ их родителями; вовлечения в дополнительное образование, в частности в деятельность Школьного Кванториума как можно больше обучающихся школ г. Ярославля, и для того, чтобы сделать доступной информацию о деятельности нашего Кванториума, а также повышения

удобства образовательного процесса для уже имеющих обучающихся. Кроме того, онлайн-сервис позволяет любому желающему получить необходимую информацию в любое удобное время.

Для вышеуказанных целей была подобрана такая технология, как чат-бот.

Чат-бот это удобный инструмент, который решает часть вопросов, связанных с коммуникацией сотрудников и учащихся. Эту задачу могут решать и другие сервисы, к примеру: мобильное приложение, сайт, онлайн-чат, горячая линия для звонков. Однако, такие подходы решают проблему менее эффективно и более энергозатратно.

Организация, рассмотренная в данной работе – Детский технопарк «Кванториум» на базе МОУ «Гимназия №3».

После внедрения сервиса у пользователей появится возможность более оперативно получать ответы на возникающие вопросы, а специалисты службы технической поддержки больше не будут тратить время на консультирование пользователей по типовым вопросам, этот процесс будет автоматизирован.

Цель: создать и ввести в эксплуатацию чат-бот в мессенджере «Telegram», который позволит обучающимся Школьного Кванториума получать сведения об образовательной организации, её новости и объявления, расписание занятий в формате более удобном, чем могут это делать сейчас.

Задачи:

- Изучить способы и алгоритмы создания Telegram-ботов
- Выявить наиболее подходящий алгоритм создания бота
- Подготовка необходимого программного обеспечения для создания бота и разработка программного кода
- Beta-test продукта проектной работы обучающимися и педагогами Кванториума и введение бота в эксплуатацию

Объект исследования – коммуникация между участниками образовательного процесса Детского технопарка «Кванториум» на базе МОУ «Гимназия №3».

Предмет исследования – чат-бот в мессенджере «Telegram».

Методы исследования: опрос обучающихся МОУ «Гимназия №3»; анализ способов и алгоритмов создания Telegram-ботов, сравнение площадок, способов и алгоритмов создания.

Новизна работы заключается в использовании современных технологий для упрощения и автоматизации некоторых процессов, о которых говорилось выше.

Личный вклад авторов работы заключается в создании продукта: Чат-бот «Телеграм».

1. Анализ способов оптимизации поиска информации

В настоящее время развивается информационный поиск, так как из большого количества данных необходимо находить только ту информацию, в которой заинтересован пользователь. Одним из способов улучшения качества поиска является использование методов анализа контента интернет-источников на основе знаний[2].

Для поиска с учетом семантики запроса существуют специальные поисковые системы. Функционал данных продуктов варьируется, начиная от переранжирования результатов поиска других поисковых систем и до полнотекстового поиска информации в проиндексированных текстах по ключевым запросам пользователей с учетом морфологических особенностей, синтаксиса и семантики слов. Так, система «Нигма» представляет собой метапоисковую систему, обеспечивающую поиск текстовой информации с учетом смысла запроса, заданного на естественном языке. Данный сервис также поддерживает формирование списка документов, разделенного на

несколько кластеров, чтобы пользователь мог уточнить, в каком кластере продолжить поиск, тем самым улучшив качество поиска. Также существуют поисковые системы, основанные на тематическом кластерном анализе, например «Carrot2». Данная система предлагает два специализированных алгоритма кластеризации: Lingo - алгоритм, основанный на сингулярном разложении, и STC - метод суффиксных деревьев. Поиск в системе «Yipru» основан на IBM Watson - суперкомпьютере, у которого есть технологии обработки естественного языка и который способен анализировать сложные, неструктурированные данные и даже понимать профессиональный сленг. «Nakia» содержит свою лингвистическую базу данных, в которой слова подразделяются на различные «смыслы», которые они передают. Она извлекает все возможные запросы, относящиеся к контенту (используя свою базу данных), и они становятся путями к исходному документу. И далее независимо ранжирует контент на основе дополнительного анализа предложений. Также для определения релевантности использует достоверность и возраст контента[3].

Несмотря на то, что множество из приведенных систем учитывают семантические особенности текста, они не используют предварительную тематическую кластеризацию и не учитывают жанровые особенности индексируемых веб-ресурсов. Также следует отметить, что большинство таких систем ориентировано на работу с текстами на английском языке[5].

Методы информационного поиска сегодня используются в разных приложениях, и одно из самых популярных приложений - мессенджер со встроенным интеллектуальным помощником или чат-ботом.

В данном проекте мы остановили свой выбор на создании чат-бота. Этот выбор обусловлен относительно простым поддержанием работоспособности и актуальности бота, возможность внедрения продукта в использование, как официальный сервис образовательной организации.

2. Выбор и описание базового варианта

Чат-боты решают две большие группы задач:

- Задачи, связанные с коммерцией. В перспективе, в Школьном Кванториуме появятся и платные образовательные услуги. Чат-бот может выступать помощником, который сможет заниматься рекламой самого себя, то есть привлекать потенциальных клиентов.
- Снижение нагрузки на службу поддержки и, следовательно, сокращение затрат. Такие алгоритмы отвечают на типовые вопросы и помогают пользователям решить их проблемы. Чаще встречаются в крупных компаниях. Соответственно, в образовательной системе можно проследить похожие задачи: получение информации об расписании, новостях и объявлениях.

Существует 2 основных алгоритма создания чат ботов:

1. Работа с интерактивными системами блочного типа программирования. Они основаны на визуальном составлении блоков в цепи, которые выполняют задачи, предусмотренные заранее.
2. Написание программного кода вручную. Такой способ сложнее, чем работа с визуальной средой, однако он позволяет глубже погрузиться в процесс создания продукта и его принципы работы. Кроме того, изучив эти процессы на уровне, который требует этот способ, можно разработать систему, которая работает чётче и стабильнее той, которая создана блочной системой. К примеру, можно увеличить разнообразие обрабатываемых запросов, встроить систему Нейросети или полноценного Искусственного Интеллекта[4].

Нами был выбран второй вариант. Это обусловлено необходимостью постоянного обновления продукта, внедрения новых функций и исправления найденных недочётов и ошибок в системе.

3.Технология создания Telegram-бота

1 Этап. Выбор подходящего набора библиотек. На этом этапе особенных трудностей, как правило, не бывает. Для решения группы задач используется отдельная библиотека. Всё предельно ясно. Например, для работы с Excel-таблицами использована библиотека “OpenExel”, для работы с API мессенджера Telegram – telebot и т.д. (Изображение 1)

2 Этап. Ознакомление с документацией к выбранным библиотекам. На этом этапе происходит изучение потенциала библиотеки, её особенностей и тонкостей.

3 Этап. Начало работы с редактором кода. На этом этапе происходят первые попытки написания бота. Пишется первая строка, первый блок кода. И так, постепенно, получается первая функция, которая отвечает на команду «/start» от пользователя. Эта функция выводит приветственное сообщение и 2 кнопки с функциями «Расписание» и «Информация» (Изображение 2)

4 Этап. От первой функции с приветственным сообщением нужно переходить на уровень выше. Нужно добавить блок «Информация», который включает в себя шаблонный текст о Школьном Кванториуме, контактные номера телефонов и ссылки. (Изображение 3)

5 Этап. Разработка способа анализа файла с информацией. Как подытог, на этом этапе мы должны получить набросок алгоритма, который должен исследовать файл с расписанием и, на основании этого анализа, должен давать ответ на запрос пользователя о расписании.

6 Этап. Создание парсинг-алгоритма, который будет сканировать файл извлекать из него полезную для нас информацию о расписании и присваивать ей номер группы в качестве уникального идентификатора.

7 Этап. Добавление полезных функций: «антифлуд» – это защита для хостинг-сервиса от нежелательных спам-атак и «репост постов из группы во ВКонтакте» – этот алгоритм дублирует посты из официального сообщества

Школьного Кванториума в личные сообщения пользователю, который подписался на рассылку от бота. (Изображение 4,5)

8 Этап. Создание удобной виртуальной клавиатуры для поиска нужной группы и вывода её расписания.

9 Этап. Добавление возможности поиска нужной группы, методом опправки боту сообщения с номером нужной группы.

10 Этап. Тестирование и отладка.

Готово!

4. Заключение

Сложностей в техническом плане практически не возникало. Сложнее было распространить информацию о наличие такого ресурса, среди обучающихся и их родителей.

С нашей точки зрения, получать необходимую от Чат-бота информацию стало гораздо удобнее. Кроме того, результаты исследования The Ultimate Guide to Chatbots, State of Chatbots 2018 г. показали, что чат-боты имеют ряд преимуществ:

- Обеспечение сервисного обслуживания круглосуточно.

Чат-бот Школьного Кванториума сможет консультировать учеников и родителей, отвечать на их вопросы независимо от часового пояса, времени суток, выходных и праздничных дней, что существенно повышает качество обслуживания и напрямую влияет на успешность.

- Экономическая эффективность.

Школьный Кванториум сможет без особых проблем внедрить виртуального помощника, чтобы отвечать на простые вопросы посетителей, а более сложные передавать техническим специалистам для последующей личной консультации клиента.

- Сбор и обработка данных о пользователях.

Сотрудники могут анализировать цепочку взаимодействий пользователя с чат-ботом, выявлять узкие места и оптимизировать процессы. На основании выполненных действия можно сегментировать аудиторию пользователей[6].

Подводя итог, можно выделить пять основных преимуществ использования чат-ботов в Школьном Кванториуме: эффективное взаимодействия с учащимися; экономическая выгода; сбор и анализ данных о пользователях; легкость в использовании; возможность проведения опросов, сбор данных об оценке качества образовательных услуг. То есть чат-бот постоянно остается «рядом» с пользователем.

5.Словарь терминов

API (Application Programming Interface или интерфейс программирования приложений) — это совокупность инструментов и функций в виде интерфейса для создания новых приложений, благодаря которому одна программа будет взаимодействовать с другой[1].

Библиотека - сборник подпрограмм или объектов, используемых для разработки программного обеспечения (ПО). С точки зрения операционной системы (ОС) и прикладного ПО, библиотеки разделяются на динамические и статические.

Оптимизация - процесс, имеющий целью направить развитие какого-либо объекта или метода к наиболее лучшему состоянию.

Чат-бот - автоматическая система для общения с пользователями.

Список литературы

1. ГОСТ Р 56716-2015 «Проектный менеджмент. Техника сетевого планирования. Общие положения и терминология» от 17.11.2015 г. – 7 с. (дата обращения 15.04.2021). - 367 с.

2. Акулич Маргарита. Чат-боты и маркетинг. М, Издательские решения – 2018 г. (дата обращения 27.04.2021). - 150 с.

3. Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Управление внедрением информационных систем. Методология внедрения OneMethodology. (дата обращения 15.04.2021). – 512 с.

4. Репин В. Бизнес-процессы. Моделирование, внедрение, управление. – «Манн, Иванов и Фербер», Москва. 2013 (дата обращения 21.04.2021). – 513

5. Ривкин Стив. Дифференцируйся или умирай! Выживание в эпоху убийственной конкуренции / Стив Ривкин, Джек Траут [Текст]. – СПб: Питер, 2010 (дата обращения 06.05.2021). – 304 с.

6. Силич В.А., Силич М.П. Системные технологии проектирования бизнес-процессов: учеб. пособие. Томск: ТПУ, 2015 (дата обращения 21.04.2021). – 108 с.

7. АБАК-ПРЕСС медиапроекты. О компании [Электронный ресурс] – URL: <http://www.apress.ru/o-kompanii/> (дата обращения 03.04.2021).

8. Алексей Семёнов. Как использовать чат-ботов: сценарии и примеры [Электронный ресурс] – URL: <https://www.uplab.ru/blog/how-to-use-chat-bots/> (дата обращения 15.05.2021).

9. Алиса – голосовой помощник от компании Яндекс. URL: <https://yandex.ru/alice/index> (дата обращения 18.03.2021).

10. Андрей Андреев, автор ApiX-Drive. Какое будущее ожидает чат-боты. Перспективы и предсказания [Электронный ресурс] – URL: <https://apix-drive.com/ru/blog/marketing/chat-bots-future> (дата обращения 17.05.2021).

Приложения

Приложение 1

Элементы кода телеграм-бота.

```
1  # -*- coding: utf-8 -*-
2  import telebot
3  import config
4  import openpyxl
5  import time as t
6  import math
7  import datetime
8  from datetime import *
9  import threading
10 import requests
11 from requests import get
12 import configparser
13 import os
14 from telebot.types import InputMediaPhoto
15 from telebot import types
16 from functools import wraps
17 import csv
18 import random
```

Изображение 1.

```
311 @bot.message_handler(commands=['unsub'])
312 def unsubscribe(message):
313     if is_not_banned(message):
314         if warn_work(message):
315             return 0
316     else:
317         if str(message.chat.id) in joinedUsers:
318             unsub(str(message.chat.id))
319             bot.send_message(message.chat.id, "Вы отписались от рассылки!\nЧтобы подписаться заного пропишите - /start")
320         else:
321             bot.send_message(message.chat.id, "Вы не подписаны!")
322
323 @bot.message_handler(commands=['start'])
324 def welcome(message):
325     global joinedUsers
326     if is_not_banned(message):
327         if warn_work(message):
328             return 0
329     else:
330         if not str(message.chat.id) in joinedUsers:
331             joinedFile = open("joined.txt", "a")
332             joinedFile.write("\n" + str(message.chat.id))
333             joinedUsers.add(str(message.chat.id))
334             joinedFile.close()
335
336         sti = open('welcome.tgs', 'rb')
337         bot.send_sticker(message.chat.id, sti)
338
339         # keyboard
340         markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
341         item1 = types.KeyboardButton("Расписание")
342         item2 = types.KeyboardButton("Информация")
343
344         markup.add(item1, item2)
345
346     bot.send_message(message.chat.id, "Добро пожаловать, {0.first_name}!\n<b>Мы - команда Ярославского Кванториума.</b> \n Этот бот позволит:\nСвязаться с нами\nУзнать рас
347     parse_mode='html', reply_markup=markup)
```

Изображение 2.

```

364 #основной перебор всех команд
365 @bot.message_handler(content_types=['text'])
366 def lalala(message):
367     if is_not_banned(message):
368         if warn_work(message):
369             return 0
370         else:
371             if message.chat.type == 'private':
372                 if message.text == 'Расписание' or message.text.lower() == 'расписание':
373                     BackToGroupsState.clear()
374                     kvantyms = []
375                     groups.clear()
376                     for i in range(len(wb.sheetnames)):
377                         kvantyms.append(wb.sheetnames[i])
378                     bot.send_message(message.chat.id, 'Выберите Квантум', reply_markup=kvants(kvantyms))
379             elif message.text == 'Информация' or message.text.lower() == 'информация' or message.text.lower() == 'инфо':
380                 info = types.InlineKeyboardMarkup(row_width=1)
381                 button1 = types.InlineKeyboardButton("Нам ВКонтакте", url='https://vk.com/yarkvantorium')
382                 button4 = types.InlineKeyboardButton("Нам Инстаграм", url='https://www.instagram.com/yarkvantorium')
383                 button5 = types.InlineKeyboardButton("Нам Одноклассники", url='https://ok.ru/group/56138091856022')
384                 button6 = types.InlineKeyboardButton("Нам Фейсбук", url='https://www.facebook.com/yarkvantorium')
385                 button2 = types.InlineKeyboardButton("Запись на занятие", url='https://yar.pfdo.ru/app/the-navigator/organization/3866')
386                 button3 = types.InlineKeyboardButton("Мы на карте", url='https://vk.cc/c1g75Y')
387                 button7 = types.InlineKeyboardButton("Позвонить нам", callback_data='phone')
388                 info.add(button1, button4, button5, button6, button3, button2, button7)
389                 bot.send_message(message.chat.id, '«КВАНТОРИУМ» - это федеральная сеть детских технопарков, создаваемая в России в рамках новой модели детского дополнительного
390
391             else:
392                 bot.send_message(message.chat.id, 'Я не знаю что ответить 😊')

```

```

551 def get_post():
552
553     global last_post_id
554     global post
555
556     post = ''
557
558     try:
559         all_posts = []
560         response = requests.get('https://api.vk.com/method/wall.get',
561                                 params={
562                                     'access_token': service_key,
563                                     'v': version,
564                                     'domain': GROUP_ID,
565                                     'count': 2
566                                 })
567
568         data = response.json()['response']['items']
569         all_posts.extend(data)
570
571         for posta in all_posts:
572             try:
573                 posta['is_pinned'] == 1
574             except:
575                 break
576
577         if posta['id'] != int(last_post_id):
578             last_post_id = posta['id']
579
580         configWrite(path)
581         oid = posta['owner_id']
582         pid = posta['id']
583         post = f'https://vk.com/wall{oid}_{pid}'
584
585     except Exception as e:
586         print(e)
587

```

Изображение 4. Парсер постов в сообществе.

```
40 ▼ def warn_work(message):
41     global warns
42 ▼     if message.chat.id not in last_time:
43         last_time[message.chat.id] = t.time()
44         return False
45 ▼     else:
46 ▼         if (t.time() - last_time[message.chat.id]) * 1000 < delay:
47             warns = warns + 1
48 ▼             if warns > max_warns:
49                 ban(message.chat.id, 240, 'Флуд!')
50                 return True
51                 last_time[message.chat.id] = t.time()
```

Изображение 5. Функция «Антифлуд».