

Сопроводительная документация к проекту  
«Умный цифровой помощник Главы региона»  
Созданный: команда Вектор

Для определения типа вопроса и его смысла мы реализовали ф-цию **get\_response (приложение 1)**, которая: разбивает введенную строку на слова, проверяет данный массив слов на схожесть с массивами **user\_input(файл bot.json – приложение 2)**. По итогу, словарь, в котором находится больше всего схожих слов **user\_input** и возвращает **bot\_response**. Если же таких сходств нет, то возвращает случайный ответ из **random\_response.json (приложение 3)**.

Само заполнение **bot.json** осуществлялось с помощью файла **inputDataInBot.py(приложение 4)**, где берется строка вопроса и желаемого ответа, разбивается по словам и с помощью шаблона **q** перезаписывается в **bot,json**.

За основу работы бота (файл **bot.py – приложение 5**) была взята библиотека **vk\_api (python)**.

В функции **main** осуществляться основная проверка событий, который приходят с **longpoll** сервера бота, с помощью цикла **for** и прослушки сервера (**bot\_longpoll.listen()**). В самом блоке цикла имеются проверки на первичный старт бота от лица пользователя и дальнейшее описание взаимодействия с ним.

Функция **question** отвечает за обработку вопросов к администрации/главы региона со стороны пользователя.

Основной функц-л бота описан в файле **botlib.py**, где находятся основные ф-ции и объекты для его дальнейшего масштабирования. Также в боте предусмотрено ведение логов в файл **BotLog.txt (приложение 6)**.

Web api организован на базе трех основных окон, с возможностью расширения под будущие задачи:

- Dashboard Окно для сбора информации о работе бота, количестве пользователей и просмотра последних поступивших запросов. (приложение )
- Newsletter Форма для отправки сообщений, с возможность прикреплять файлы, всем пользователям, например, произошло обновление документов и нужно сообщить всем об этом, это окно поможет это сделать.

- Settings Окно для редактирования (создания / удаления / изменения / прикрепления файлов) запросов и ответов на них, также с редактированием ключевых слов по которым, определяется наиболее подходящий ответ для пользователя. В разработке Web api использовалась базовая верстка на html с использованием таблицы каскадных стилей. Обработка действий на странице реализована на JavaScript. Данный стек технологий позволяет легко и быстро модифицировать продукт под новые нужды.

## Приложение 1

```
def get_response(input_string):
    split_message = re.split(r'\s+|[,;?!.-]\s*', input_string.lower())
    score_list = []

    # Check all the responses
    for response in response_data:
        response_score = 0
        required_score = 0
        required_words = response["required_words"]

        # Check if there are any required words
        if required_words:
            for word in split_message:
                if word in required_words:
                    required_score += 1

        # Amount of required words should match the required score
        if required_score == len(required_words):
            # print(required_score == len(required_words))
            # Check each word the user has typed
            for word in split_message:
                # If the word is in the response, add to the score
                if word in response["user_input"]:
                    response_score += 1

        # Add score to list
        score_list.append(response_score)
        # Debugging: Find the best phrase
        # print(response_score, response["user_input"])

    # Find the best response and return it if they're not all 0
    best_response = max(score_list)
    response_index = score_list.index(best_response)

    # Check if input is empty
    if input_string == "":
        return "Please type something so we can chat :("

    # If there is no good response, return a random one.
    if best_response != 0:
        return response_data[response_index]

    return random_responses.random_string()
```

## Приложение 2

```

{
  "response_type": "question",
  "user_input": [
    "за",
    "магазином",
    "восток",
    "по",
    "кирова",
    "ежегодно",
    "собирается",
    "большая",
    "лука",
    "стока",
    "воды",
    "нет",
    "уходит",
    "под",
    "дома",
    "боимся",
    "размытия",
    "связи",
    "близ",
    "лежащих",
    "домов. 2"
  ],
  "bot_response": "Сейчас пока ревеня нет, в связи с отсутствием финансирования на производство по организации стока воды- водоотводных лотков, поднятие полотна бетонного и тп. Техни",
  "required_words": []
},
{
  "response_type": "question",
  "user_input": [
    "продолжится",
    "ли",
    "озеленение",
    "насыпей",
    "(искусственных",
    "гор)",
    "в",
    "близ",
    "города.",
    "Мне",
    "кажется",
    "что",
    "это",
  ]
}

```

### Приложение 3

```

import random

def random_string():
    random_list = [
        "Пожалуйста, попробуйте написать что-нибудь более описательное.",
        "Ой! Кажется, ты написал что-то, чего я пока не понимаю",
        "Вы не против попробовать перефразировать это?",
        "Мне очень жаль, я не совсем уловил это.",
        "Я пока не могу ответить на этот вопрос, попробуйте спросить что-нибудь еще."
    ]

    list_count = len(random_list)
    random_item = random.randrange(list_count)

    return random_list[random_item]

```

### Приложение 4

```

#Алгоритм для ввода данных в bot.json

import json

uinput='Ваш вопрос'
bresponse="" "Желаемый ответ бота""

#Шаблон для заполнения
q={
    "response_type": "question",
    "user_input": [],
    "bot_response": "",
    "required_words": []
}

#Чтение файла, добавление нового словаря
with open ('bot.json','r',encoding='utf-8') as bot:
    data=json.load(bot)

uinput=uinput.lower()

q['user_input']=uinput.split()
q['bot_response']=bresponse
data.append(q)

#Перезапись исходника (с добавлением новых данных)
with open ('bot.json','w',encoding='utf-8') as bot:
    bot.write(json.dumps(data,ensure_ascii=False,indent=2))

```

## Приложение 5

```

#Функция для обработки вопросов
def question(event:VkBotMessageEvent): --

def main():
    print("BOT STARTED")

    print('Launch scenary: ', main.__name__)

    #Прослушивание и обработка событий с сервера
    for event in botLib.bot_longpoll.listen():

        if event.type==VkBotEventType.MESSAGE_NEW and event.from_user and event.message['text']=='Начать':
            botLib.writeInFile_start(event.message['from_id'],event.message['from_id'],'user')
            botLib.session.method('messages.send',{
                'user_id': event.message['from_id'],
                'random_id': random.randint(1,1000)+time.localtime().tm_sec,
                'attachment': 'photo-223836799_456239819',
                'message': 'Привет! Это команда [Vector]. Сейчас у нас идет тестирование кейса [Хакатон: Лидеры Цифровой трансформации].\nЕсли вы хотите по взаимодействовать с ботом по это
            })

        #Обработка основных сообщений для бота
        elif event.type==VkBotEventType.MESSAGE_NEW and event.from_user:
            botLib.session.method('messages.send',{
                'user_id': event.message['from_id'],
                'random_id': random.randint(2,1000)+time.localtime().tm_sec,
                'message': 'Обработка запроса'
            })

        if (event.message['text']!=None or event.message['text']!=' '):

            #Генерация ответа от бота
            answer=get_response(event.message['text'])

            #Отправка ответа ботом
            if type(answer) !=str and answer['response_type']=='greeting':
                botLib.session.method('messages.send',{
                    'user_id': event.message['from_id'],
                    'random_id': random.randint(3,1000)+time.localtime().tm_sec,
                    'keyboard': botLib.keyboard_start.get_keyboard(),
                    'message': answer['bot_response'][:len(answer['bot_response'])-2]
                })

```

## Приложение 6

104	[LOG]	[2023-12-17 12:20:39.695591]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
105	[LOG]	[2023-12-17 12:22:19.735776]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
106	[LOG]	[2023-12-17 12:22:24.194330]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
107	[LOG]	[2023-12-17 12:23:28.549927]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: хай
108	[LOG]	[2023-12-17 12:23:32.919140]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
109	[LOG]	[2023-12-17 12:25:34.833374]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Будут ли строить новые гаражи?
110	[LOG]	[2023-12-17 12:26:37.434381]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Будут ли строить новые гаражи?
111	[LOG]	[2023-12-17 12:27:37.848158]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Будут ли строить новые гаражи?
112	[LOG]	[2023-12-17 12:30:22.863481]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
113	[LOG]	[2023-12-17 12:30:33.823792]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
114	[LOG]	[2023-12-17 12:30:38.798218]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
115	[LOG]	[2023-12-17 12:30:43.674534]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
116	[LOG]	[2023-12-17 12:30:45.407701]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
117	[LOG]	[2023-12-17 12:30:48.125519]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
118	[LOG]	[2023-12-17 12:31:05.035180]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
119	[LOG]	[2023-12-17 12:31:10.018675]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
120	[LOG]	[2023-12-17 12:31:15.288048]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
121	[LOG]	[2023-12-17 12:34:23.388712]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Хай
122	[LOG]	[2023-12-17 12:36:14.177195]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Хай
123	[LOG]	[2023-12-17 12:36:43.879614]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Что делт
124	[LOG]	[2023-12-17 12:36:59.808572]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
125	[LOG]	[2023-12-17 12:40:54.818281]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['back']
126	[LOG]	[2023-12-17 12:40:55.638143]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['back']
127	[LOG]	[2023-12-17 12:42:02.110765]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Ну шо
128	[LOG]	[2023-12-17 12:42:21.463614]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Гацдон ты, все нервы из меня вывел
129	[LOG]	[2023-12-17 12:42:28.668794]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Ахахх
130	[LOG]	[2023-12-17 12:50:52.868967]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Гацдон ты, все нервы из меня вывел
131	[LOG]	[2023-12-17 12:52:10.581845]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Гацдон ты, все нервы из меня вывел
132	[LOG]	[2023-12-17 12:52:15.555892]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как дела?
133	[LOG]	[2023-12-17 12:52:20.464082]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как дела
134	[LOG]	[2023-12-17 12:52:25.617126]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как ты
135	[LOG]	[2023-12-17 12:53:11.994611]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как дела
136	[LOG]	[2023-12-17 12:53:16.984865]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как ты
137	[LOG]	[2023-12-17 17:31:38.397936]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Как узнавать какие соревнования заложены на год, а то по факту приходишь, а
138	[LOG]	[2023-12-17 17:31:44.742020]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
139	[LOG]	[2023-12-17 17:31:52.552692]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
140	[LOG]	[2023-12-17 20:24:14.525914]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Хоба
141	[LOG]	[2023-12-17 20:24:19.116030]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Начать
142	[LOG]	[2023-12-17 20:25:06.559643]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Начать
143	[LOG]	[2023-12-17 20:25:12.028001]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_NEW	Text: Привет
144	[LOG]	[2023-12-17 20:25:16.786931]	UserID: 170852963	Scenario: start	Event: VkBotEventType.MESSAGE_EVENT	Payload: ['question']
145						