

# **Отчет по лабораторной работе №4**

Данила Андреевич Стариков

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>3</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>4</b>
<b>3</b>	<b>Выводы</b>	<b>12</b>

# 1 Цель работы

Составить JSON схему, описывающую все возможности метода из лабораторной работы №3 (WEB API Вконтакте). JSON схема должна быть составлена и для объекта, который посылается в запросе и для объекта, который приходит в ответе от сервера. Проверить правильность схемы проверьте с помощью валидатора, например .

## 2 Выполнение лабораторной работы

Будем составлять JSON схему для метода docs.get. Сначала покажем пример работы этого метода:

```
http POST https://api.vk.com/method/docs.get \
access_token==vk1.a.ZdjvFQIhBuM-HEhyBotNorKjYeu_rjgevnostJtKePu5UuGLz4z8_
↪ mz006GKmrM_Kk-eA5E8qsKjJbANnt2XU6_N-Zh5uNZGjHndgbdN_IftuPuxau-Byr_jP_
↪ x9zB5Z02vXzvTVx27wt5Eg-Da8YH4ZliMHBI0ppoSeF2oBd4QedWd0HJ-tBeBRESAQKW_
↪ 22AfdjF1d8kbzaoB2q-CW9Kgeg
↪ \
v==5.199 \
count==5
```

Ответ:

```
{
  "response": {
    "count": 329,
    "items": [
      {
        "can_manage": true,
        "date": 1702456522,
        "ext": "pdf",
        "id": 670812375,
        "is_unsafe": 0,
```

```

    "owner_id": 85081472,
    "size": 1903102,
    "title": "Гамильтоновы циклы .pdf",
    "type": 1,
    "url": "https://vk.com/doc85081472_670812375?hash=hS4lZ4_
    ↪ ZBWw7Ncbz4ilLTZB2lxIBMZNiSH6UGrONxbGz&dl=a7TsTAQKlhE_
    ↪ KhDKYG7HHXzN9pGAKvWoIdJKpVR8VtUX&api=1&no_preview=1"
  },
  {
    "can_manage": true,
    "date": 1678000031,
    "ext": "pdf",
    "id": 657272932,
    "is_unsafe": 0,
    "owner_id": 85081472,
    "size": 2152747,
    "title": "Домашнее задание 3.pdf",
    "type": 1,
    "url": "https://vk.com/doc85081472_657272932?hash=dqvgB5_
    ↪ Xq26A98155W9ThY3DTJTqXmrLviQsjnoZpfq8&dl=c1CYKNAZ7sj_
    ↪ XYZQWuo03Fbrcdkgm4vsndPhQuKbsJ30&api=1&no_preview=1"
  },
  {
    "can_manage": true,
    "date": 1672062726,
    "ext": "pdf",
    "id": 653151850,
    "is_unsafe": 0,
    "owner_id": 85081472,

```

```

    "size": 4117280,
    "title": "Подготовка к зачету_221226_144325.pdf",
    "type": 1,
    "url": "https://vk.com/doc85081472_653151850?hash=KF5plXj
    ↪ D4Gec5C5Xwfcz5zwia0zH5DMUUCWhAhHDFxOk&dl=grM0dtDiXgfj
    ↪ PJWca8mdDpedQf9CCfs58pbvMk7X4q5o&api=1&no_preview=1"
  },
  {
    "can_manage": true,
    "date": 1671884548,
    "ext": "pdf",
    "id": 653011190,
    "is_unsafe": 0,
    "owner_id": 85081472,
    "size": 2515804,
    "title": "Notes_221224_152122.pdf",
    "type": 1,
    "url": "https://vk.com/doc85081472_653011190?hash=Zuzwpsj
    ↪ ckqWrEOXipCh2ep0b9lrNjry38pYxBZ2jYriH&dl=qfTk5Mjex9j
    ↪ 3tOJYwKCkQ0GVtVvZqPN0vMZTFxKFh8P&api=1&no_preview=1"
  },
  {
    "can_manage": true,
    "date": 1671535324,
    "ext": "pdf",
    "id": 652721733,
    "is_unsafe": 0,
    "owner_id": 85081472,
    "size": 8594963,

```

```

        "title": "Подготовка к кр_221207_082207.pdf",
        "type": 1,
        "url": "https://vk.com/doc85081472_652721733?hash=Seblnf_
        ↪ 8He5kCbmEhTGZ0j2r51IoGs9mYhIR3UgzRV10&dl=h6LDxvWYnUo_
        ↪ ogJxsQ3CwF9IqvFgzkKL VX8Q0znhJMz4&api=1&no_preview=1"
    }
]
}
}

```

Теперь запишем JSON схему для запроса к этому методу:

```

{
    "name": "docs.get",
    "description": "Возвращает расширенную информацию о документах
    ↪ пользователя или сообщества.",
    "access_token_type": [
        "user"
    ],
    "parameters": [
        {
            "name": "count",
            "description": "Количество документов, информацию о которых
            ↪ нужно вернуть. По умолчанию: все документы. Максимальное
            ↪ количество документов, которое можно получить: 2000.",
            "type": "integer",
            "maxItems": 2000,
            "minimum": 1
        },
        {

```

```

    "name": "offset",
    "description": "Смещение, необходимое для выборки
↪ определенного подмножества документов. Максимальное
↪ значение: 1999",
    "type": "integer",
    "maxItems": 1999,
    "minimum": 1
  },
  {
    "name": "type",
    "description": "Тип файла. Возможные значения:",
    "type": "integer",
    "minimum": 1,
    "maximum": 8
  },
  {
    "name": "owner_id",
    "description": "Идентификатор пользователя или сообщества,
↪ которому принадлежат документы.",
    "type": "integer"
  },
  {
    "name": "return_tags",
    "description": "",
    "type": "integer",
    "minimum": 0,
    "maximum": 1
  },
],

```



```

    "responses": {
      "response": {
        "$ref": "#/docs_get_response"
      }
    }
  }
}

```

Опишем структуру объекта, приходящего в ответ:

```

{
  "docs_get_response": {
    "type": "object",
    "properties": {
      "response": {
        "type": "object",
        "properties": {
          "count": {
            "type": "integer",
            "description": "Количество элементов в массиве \"items\"",
            "minimum": 0
          },
          "items": {
            "type": "array",
            "description": "Массив объектов файлов",
            "items": {
              "$ref": "#/definitions/object_file"
            }
          }
        }
      }
    }
  },
  "required": [
    "response"
  ],
  "additionalProperties": false
}

```

Запишем JSON схему для объекта файла:

```

{
  "object_file": {
    "type": "object",
    "description": "Объект, описывающий файл",
    "properties": {
      "id": {
        "type": "integer",
        "description": "Идентификатор файла"
      },
      "owner_id": {
        "type": "integer",
        "description": "Идентификатор пользователя, загрузившего файл"
      }
    }
  },

```

```

"title": {
  "type": "string",
  "description": "Название файла"
},
"size": {
  "type": "integer",
  "description": "Размер файла в байтах"
},
"ext": {
  "type": "string",
  "description": "Расширение файла"
},
"url": {
  "type": "string",
  "description": "Адрес файла, по которому его можно загрузить"
},
"date": {
  "type": "integer",
  "description": "Дата добавления в формате Unixtime"
},
"type": {
  "type": "integer",
  "description": "Тип файла. Возможные значения: 1 — текстовые документы; 2 — архивы; 3 — gif; 4 — изображения; 5 — аудио; 6 — видео; 7  
↔ — электронные книги; 8 — неизвестно.",
  "minimum": 1,
  "maximum": 8
},
"preview": {
  "type": "object",
  "description": "Информация для предварительного просмотра файла.",
  "properties": {
    "photo": {
      "type": "object",
      "properties": {
        "sizes": {
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "src": {
                "type": "string"
              },
              "width": {
                "type": "integer",
                "minimum": 0
              },
              "height": {
                "type": "integer",
                "minimum": 0
              },
              "type": {
                "enum": ["s", "m", "x", "o", "p", "q", "l", "y", "z", "w"]
              }
            }
          }
        }
      }
    }
  }
},
"graffiti": {
  "type": "object",

```

```

        "properties": {
          "src": {
            "type": "string"
          },
          "width": {
            "type": "integer",
            "minimum": 0
          },
          "height": {
            "type": "integer",
            "minimum": 0
          }
        }
      },
      "audio_message": {
        "type": "object",
        "properties": {
          "duration": {
            "type": "integer"
          },
          "waveform": {
            "type": "array",
            "items": {
              "type": "integer"
            }
          },
          "link_ogg": {
            "type": "string"
          },
          "link_mp3": {
            "type": "string"
          }
        }
      }
    }
  }
}

```

Для проверки созданной JSON схемы создали два файла: docs\_get\_response\_schema.json, где описана JSON схема, и docs\_get\_response.json, куда записан ответ, пришедший при выполнении следующего запроса:

```

http POST https://api.vk.com/method/docs.get \
  access_token=vk1.a.0C164Tbc6nX0zpIPxmFW0iK60H1N08C2QmttWJDBRbUzGn7BIfs6n50TcP59JvBtsGNKLgv4z1hWXX8NUGewQ6p9Vt7UIu6p4Lj0x561kybTwVXPMuFjtHdX6Is_Ej
  ↪ iqbZB1Z8tdfFns8TKix0xsvzYDyamJWVSeWLy0sqWvQ9cN59H5V6hbQ7Ktp1UYev5FY6X-60XLwjV1qi7sF0HwJYw
  ↪ \
count==25 \
v==5.199 \
-o docs_get_response.json

```

Далее с помощью пакета jsonschema в командной строке запустили проверку:

```
jsonschema -i docs_get_response.json docs_get_response_schema.json -o pretty
```

```

[ 8:26PM ] [ dastarikov@dastarikov:~/work/study/2024-2025/web-services-course/ ]
$ jsonschema -i docs_get_response.json docs_get_response_schema.json -o pretty
===[SUCCESS]===(docs_get_response.json)===

```

Рис. 2.1: Результат проверки ответа сервера с помощью созданной JSON схемы

## 3 Выводы

В ходе выполнения лабораторной работы разработали JSON схему по стандарту JSON Schema на основе метода `docs.get` API Вконтакте.