

# Руководство пользователя программного продукта СИП "Цифровая приемная"

---

## 1. Предисловие

Сервис СИП "Цифровая приемная" компании Ред Софт предназначен для распознавания входящих документов (напр. обращений граждан или организаций) и для автоматизации регистрации таких документов в системах АИС. Обращения могут быть в формате ГОСТ Р 7.0.97-2016, так и в нестандартной форме.

Система относится к решениям на основе технологии искусственного интеллекта в области управления бизнес-процессами и предназначена для автоматического назначения исполнителя и вынесения резолюции на основе текста входящего обращения и другой информации из обращения. Если обращение поступило в виде скан-копии, в форматах .pdf, .png, .jpg, .jpeg, .bmp, СИП "Цифровая приемная" распознаёт блоки полей: Отправитель, Информация об отправителе, Получатель, Номер документа, Тема, Основной текст, Должность, Подпись, ФИО подписанта и т.д, что исключает ввод необходимой для назначения исполнителя и вынесения резолюции информации вручную.

В данном руководстве описана версия ПО 0.2.16-2.

Система представляет собой веб сервис. Дистрибутив системы распространяется посредством системы контейнеризации docker.

Сервис использует протокол HTTP для обмена.

## 2. API

Поскольку система выполнена в виде веб сервиса, далее описываем методы сервиса.

### 2.1. Получение версии сервиса

Предназначен для получения текущей версии сервиса, а также версий компонент входящих в состав сервиса.

Url: /api/v1/info

Тип запроса: *GET*

Параметры: без параметров

Пример:

```
curl http://localhost:8080/api/v1/info
```

Ответ: во всех случаях ответа сервиса предполагается формат ответа JSON

```
{
  "data":
  {
    "DocIA Service": "0.2.16",
    "pdfium": "108.0.5323",
    "tesseract": "5.2.0",
    "onnxruntime": "1.12.1"
  }
}
```

## 2.2. Запрос на распознавание документа/тов.

Основной запрос сервиса, предназначен для распознавания графического образа документа в формате pdf, png, jpg.

Url: /api/v1/run

Тип запроса: *POST*, Content-Type - multipart/form-data

Параметры: произвольное число параметров с любыми не повторяющимися именами файлов. В качестве вложений запроса, должны быть файлы с изображениями документов. Имя файла вложения также должно включать поддерживаемое расширение файла *pdf*, *jpg* или *png*.

Пример:

```
curl -F 'doc1=@./Scan2021040810105644700001.pdf' -F 'doc2=@./big.pdf'
http://localhost:8080/api/v1/run
```

Ответ:

В случае успешного ответа сервиса, ответ представлен в виде JSON по каждому документу запроса в виде блоков информации о местоположении каждого блока, его типе и тексте внутри блока. Все координаты блоков представлены в относительных величинах относительно размеров самого документа в интервале [0-1].

```
{
  "files": {
    "doc1": {
      "blocks": [
        {
          "id": "title_long",
          "scores": {
            "ns1": 0.73687685
          },
          "rect": {
            "x": 0,
            "y": 0,
            "w": 1,
```

```
        "h": 0.8978201634877384
      },
      "page": 0
    },
    {
      "id": "from_info",
      "text": "В Государственную Думу Федерального Собрания
Российской ...",
      "scores": {
        "ocr": 0.94,
        "ns3": 0.9924259,
        "ns2": 0.943655
      },
      "rect": {
        "x": 0.1383132530120482,
        "y": 0.5616485013623979,
        "w": 0.7537349397590362,
        "h": 0.3361716621253406
      },
      "page": 0
    },
    {
      "id": "barcode",
      "scores": {
        "ns2": 0.9386001
      },
      "rect": {
        "x": 0.6809638554216867,
        "y": 0.08412806539509536,
        "w": 0.22457831325301206,
        "h": 0.07970027247956403
      },
      "page": 0
    },
    {
      "id": "from",
      "text": "107996, г. Москва, ул. Кузнецкий Мост, д. 1,
стр. 7",
      "scores": {
        "ns2": 0.9263059,
        "ns3": 0.998754,
        "ocr": 0.94
      },
      "rect": {
        "x": 0.5604819277108434,
        "y": 0.4720708446866485,
        "w": 0.2433734939759036,
        "h": 0.03746594005449591
      },
      "page": 0
    },
    {
      "id": "from",
      "text": "Директору ... главному ...",
```

```
        "scores": {
            "ns2": 0.9145153,
            "ocr": 0.94,
            "ns3": 0.99989855
        },
        "rect": {
            "x": 0.5633734939759036,
            "y": 0.3739782016348774,
            "w": 0.31710843373493974,
            "h": 0.0752724795640327
        },
        "page": 0
    },
    ...
],
"pages": 5
},
"doc2": {
    "blocks": [
        ...
    ],
    "pages": 5
}
}
```

### 3. Обработка ошибок

В случае если происходит ошибка, ответ также возвращается в формате JSON.

В данном примере, запрос происходит на несуществующий url:

```
$ curl http://localhost:8080/api/v1/info1
{
  "data":null,
  "errors":
  [
    {
      "message":"Not found resource",
      "code":404,
      "detail":"/api/v1/info1"
    }
  ]
}
```

Ответ при этом содержит поле *errors* со списком ошибок.