

# VoidShelf

## Инструкция по эксплуатации

1. Создать задание на обработку скриншотов, используя запрос API п. 2. В запросе требуется указать идентификатор расписания ЕЦХД. В случае успешного выполнения запроса, в ответе на запрос Вы получите идентификатор созданной задачи.
2. Дождаться старта выполнения задания. В момент, когда система сможет получить скриншоты, статус задачи изменится на `downloading`. Статус задачи Вы сможете узнать, сделав запрос на статус задачи согласно 4 пункту API.
3. Когда система загрузит все скриншоты, статус задачи изменится на `inProgress`. В шлюз `rabbitmq` начнут поступать сообщения с результатами обработки по каждому скриншоту.
4. После завершения отработки, статус задачи изменится на `done`. Если расписание периодическое - задача снова начнет обработку при получении новых скриншотов.

# HTTP API

## 1. Актуализация списка расписаний ЕЦХД

GET http://addr:port/schedules/refresh

Ответ:

```
{"msg": "OK"}
```

## 2. Создание задачи на обработку

POST http://addr:port/tasks/create

Тело запроса (пример):

```
{
  "detectors": [5],
  "schedule": "0ba05dac-62fb-4ef3-bf93-1d7e187776d6"
}
```

Где

detectors - типы детекторов (Integer),

schedule - идентификатор расписания ЕЦХД.

Ответ:

```
{
  "msg": "OK",
  "taskId": "275bf358-6948-4f3b-af7c-81eb1242908f",
}
```

где

msg - сообщение об успешном создании или ошибке,

taskId - идентификатор задачи (UUID4)

## 3. Список созданных заданий

GET http://addr:port/tasks/index

Ответ:

```
{"tasks": [
  {"id": "85a4bbda-f3b3-4b89-8693-ee456a790019", "status": "InProgress"},
  {"id": "fb8f3884-4202-4df8-8d9b-9344217bf210", "status": "paused"},
  ...
]}
```

## 4. Получение статуса

GET http://addr:port/tasks/taskId/status

Ответ:

```
{
  "detectors": [5],
  "started": 1558351502,
  "approximateDuration": 3600,
  "status": "InProgress",
  "percentage": 57.0,
  "schedules": [
    "0ba05dac-62fb-4ef3-bf93-1d7e187776d6",
    "1aa7f07c-ccc0-4949-af51-b1901b54fb16"
  ]
}
```

где

detectors - типы детекторов (Integer)

approximateDuration - расчетное время окончания в секундах (Integer)

started - время начала выполнения задачи (UNIX)

status - статус задачи (String):

- started - задача создана и началось ее выполнение
- downloading - скачивание скриншотов для анализа
- inProgress - происходит анализ скриншотов
- paused - задача приостановлена
- canceled - задача отменена
- done - задача выполнена
- failed - задача завершилась с ошибкой

При отсутствии останова/ошибок смена статусов имеет следующий характер:

started > downloading > inProgress > done > started > ...

percentage - процент выполнения (Double)

schedules - идентификаторы расписаний ЕЦХД

## 5. Получение результатов анализа по задаче

GET http://addr:port/results/taskId

Ответ:

```
{
  "results": [
    {
      "screenshotId": "screenshot1",
      "label": 51,
      "score": 98
    },
    {
      "screenshotId": "screenshot2",
      "label": 50,
      ...
    }
  ]
}
```

где

screenshotId - идентификатор скриншота (String)

label - тип нарушения (Integer)

score - степень уверенности

Параметры запроса:

from - все результаты начиная с указанного времени (UNIX)

to - все результаты до указанного времени (UNIX)

Пример: http://addr/port/results/275bf358-6948-4f3b-af7c-81eb1242908f?  
from=1558351300&to=1558353000

## 6. Приостановить задачу

PATCH http://addr:port/tasks/taskId/pause

Ответ:

```
{"msg": "OK"}
```

## 7. Возобновить задачу

PATCH http://addr:port/tasks/taskId/resume

Ответ:

```
{"msg": "OK"}
```

## 8. Отменить задачу

PATCH <http://addr:port/tasks/taskId/cancel>

Ответ:

```
{"msg": "OK"}
```

## 9. Формат сообщения, передаваемого в шлюз:

```
{
  "results": [
    {
      "screenshotId": "a87189cd-766c-4992-b04e-5339fd161a68",
      "taskId": "2c838098-f6b1-4ed7-9c54-9d2240e57dbb",
      "label": 51,
      "score": 98,
    }
  ],
  "extras": {
    "totalCount": 10,
    "totalArea": 240,
  }
}
```

где

screenshotId - идентификатор скриншота (String)

label - тип нарушения (Integer)

score - степень уверенности (Integer)

extras.totalCount - общее количество пустых полок (Integer)

extras.totalArea - суммарная площадь пустых полок (Integer)