



## **Face PSIM. REST API**

Face PSIM 2.0 (english)

Last update 09/17/2025

## Table of Contents

1	General information .....	6
2	Get the recognition protocol .....	7
2.1	General request format:.....	7
2.2	Request parameters: .....	7
2.3	Request example: .....	8
2.4	Request headlines:.....	8
2.5	Response example:.....	9
2.6	Response parameters:.....	10
3	Add/update a user in the face database .....	13
3.1	General request format:.....	13
3.2	Request parameters: .....	13
3.3	Request example: .....	14
3.4	Request headlines:.....	19
3.5	Response example:.....	20
3.6	Response parameters:.....	20
4	Update user in the face database .....	21
4.1	General request format:.....	21
4.2	Request parameters: .....	21
4.3	Request example: .....	22
4.4	Response headlines:.....	22
4.5	Response example:.....	23
4.6	Response parameters:.....	23
5	Get information about users from a specified department or the entire face database.....	24
5.1	General request format:.....	24
5.2	Request parameters: .....	24
5.3	Request example: .....	24

5.4	Request headlines:.....	25
5.5	Response example:.....	26
5.6	Response parameters:.....	26
6	Delete users from the face database .....	28
6.1	General request format:.....	28
6.2	Request parameters: .....	28
6.3	Request example: .....	28
6.4	Request headlines:.....	29
6.5	Response example:.....	29
6.6	Response parameters:.....	29
7	Search for faces by image.....	31
7.1	General request format:.....	31
7.2	Request parameters: .....	31
7.3	Request example: .....	31
7.4	Request headlines:.....	35
7.5	Response example:.....	36
7.6	Response parameters:.....	37
8	Get an image by ID.....	40
8.1	General request format:.....	40
8.2	Request parameters: .....	40
8.3	Request example: .....	40
8.4	Request headlines:.....	40
8.5	Response example:.....	41
9	Get analytical data .....	42
9.1	General request format:.....	42
9.2	Request parameters: .....	42
9.3	Request example: .....	43
9.4	Request headlines:.....	44
9.5	Response example:.....	44

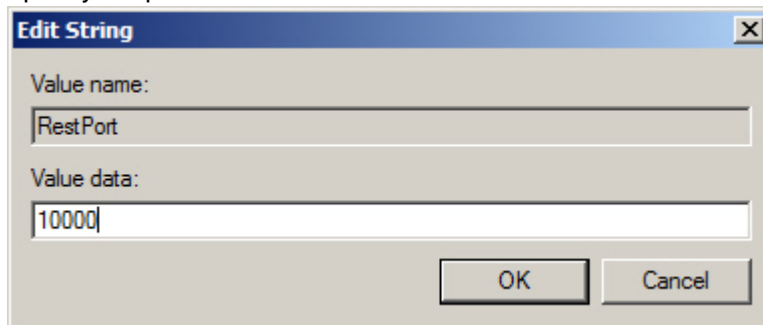
9.6	Response parameters:.....	46
10	Get analytical data by time.....	49
10.1	General request format:.....	49
10.2	Request parameters: .....	49
10.3	Request example: .....	50
10.4	Request headlines:.....	50
10.5	Response example:.....	51
10.6	Response parameters:.....	52
11	List of possible errors of the Face recognition server.....	54
11.1	General request format:.....	54
11.2	Request example: .....	54
11.3	Response:.....	54
11.4	Request headlines:.....	54
11.5	Response parameters:.....	55
12	Face search by camera and time.....	56
12.1	General request format:.....	56
12.2	Request parameters: .....	56
12.3	Request example: .....	56
12.4	Request headlines:.....	57
12.5	Response example:.....	57
12.6	Response parameters:.....	58
13	Number of faces in the frame.....	60
13.1	General request format:.....	60
13.2	Request parameters: .....	60
13.3	Request example: .....	60
13.4	Request headlines:.....	60
13.5	Response example:.....	61
13.6	Response parameters:.....	61
14	Get a list of all departments.....	62

14.1 General request format:.....	62
14.2 Request parameters: .....	62
14.3 Request example: .....	62
14.4 Request headlines:.....	62
14.5 Response example:.....	63
14.6 Response parameters:.....	63

# 1 General information

Before start of using the REST API and launch the *Face PSIM* software package, do the following:

1. Select the HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\AxxonSoft\PSIM\FRS registry section.
2. Specify the parameter value **RestPort** = 10000.



The software implementation of the *Face PSIM* HTTP API is similar to [Axxon PSIM HTTP API](#).

General agreements, limitations and features are described on the [General information on HTTP API](#) page; additional settings required for the *Face PSIM* API operation are described on the [Calling API of vertical solutions via Axxon PSIM HTTP API](#) page.

In the *Face PSIM* API, **FirServer** corresponds to the **Face Recognition Server** software module, and the default port is 10000.

## 2 Get the recognition protocol

### 2.1 General request format:

POST `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/GetProtocols`

### 2.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
onlineRefresh	Yes	This parameter must be set to 1
genders	No	Gender: 0—undefined, 1—male, 2—female If not specified, gender is not taken into account
page	No	Number of pages with results If not specified, then page = 1
pageSize	No	Number of results per page If not specified, then pageSize = 1, that is, there is one result on one page
cameraIds	No	Camera IDs (GUID or ID in the <i>Axxon PSIM</i> object tree) If not specified, then results for all cameras are displayed
sim_min	No	Face types: -1—all faces, 1—only recognized faces, 0—only unrecognized faces If not specified, then sim_min = -1
minAge	No	Minimum age of persons If not specified, age is not taken into account

Parameter	Required	Parameter description
maxAge	No	Maximum age of persons If not specified, age is not taken into account
dateTimeFrom	Yes	The initial boundary of the time period in UTC+0 format: <b>yyyy-mm-ddThh:nn:ss.fff</b>
dateTimeTo	Yes	The end boundary of the time period in UTC+0 format: <b>yyyy-mm-ddThh:nn:ss.fff</b>

## 2.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/GetProtocols

```
{
  "server_id": "1",
  "onlineRefresh": 1,
  "genders": [
    0,
    1,
    2
  ],
  "dateTimeFrom": "2020-05-14T16:25:55.687",
  "dateTimeTo": "2020-05-16T16:26:55.687",
  "cameraIds": [
    "1"
  ],
  "minAge": 0,
  "maxAge": 1000,
  "page": 1,
  "pageSize": 50,
  "sim_min": -1
}
```

## 2.4 Request headlines:

Headline	Headline description
Allow	Returns the HTTP method used in request

Headline	Headline description
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 2.5 Response example:

```
{
  "Protocols": [
    {
      "id": "18D10CC2-9596-EA11-8A9A-B06EBF8119EF",
      "camera_id": "5",
      "camera_name": "grundig 5",
      "timestamp": "2020-05-15T10:20:54.657",
      "age": 55.4815,
      "gender": 2,
      "emotion": 4,
      "glasses": 2,
      "facialHair": 6,
      "hairColor": 1,
      "hairType": 3,
      "headwear": 13,
      "liveness": 1,
      "evasion": 1,
      "temperature": 36.6,
      "fir": {
        "data": "rv0GvR00xbx76+a8kIHXPfbzkbwY7hA9vpS5PRdb0j32pg4+yqerPT6Ghjw/Vhy9PpMQvWHqmb2U5VQ8kic/vWIS/b0xXNq8kJ7hPU7MGT2qdxw9FdAJPFxwtLxGLUQ9wBJCPLPiB71UMoU8vhbHP0Thqb3VaSg92qH0PH0j1zyEda a9u10xvVfWq71oy0o8twbWu3g+mD0SQvw8C/KhvEjElrspZF19Q7xsu6FLol1mLa492n3tvCyBl7wNll8+n+6RvRQnCbzDDhw9EnCAPdSsmT2PsHa8n+6HvTd 6Cj0qT3U9oFQMveIDNj0m7sU8eih0vYGvgT3INic9tWfXvevQnr1QMmG92WGHvf7ULzxx74a8aCmlvXAa6j0v EM08ApXTvGsovLw0uzC9TpGt00eUfbzB0F4+SUgqPQPevbyr+K08NJQTve1zZLrt6kQ9lvEvZCE/"
      }
    }
  ]
}
```

```

j0R9B693z+APJK7l7x6K+W9yPTm08ai1jp630U90KUUPujXET0WnPa8DIW7PE32Z73Fkke8kYNmvZg/
Lb2bv+o95BUKvYRH47yr70S8ojApvaMX5bwC4Co9AuwNvSFoWTyaXa09ueWFvPZ+RT3w7Mk7HnsLvDsmBD32P
AW9IuphvSZLpTyBN5W9+e4sPcuAXr2RNI288I5hvYAm3LtEHnm9FFVfVio8VL0ishw+QHjDPGGbhr2sJZK8E3
a5PYsDK73kgKu8cN4/
PB3Z8z10AE+9HWm7PRJhdr5l7Tm9LjRHvZtTA7wTW+08nGa4PXmX0bs1mwk8xrGcPBgNfzyfF7A6vYPPaRxU
L3jK8e7Y3K0vX5i77z543o9Xh/9PF0eAD24hrG8VYL1vRQnhrxGpcm8uu0C0dmLAz0z2AG+0cChu6l/
Yz1Xdla9fHPnvAZuij2dv9g7hlyu47LJb1fUFM9IpIcvSRgBr7RC4Y9GgFCPeuYTLyw2oe9wUZNpZAPGr0Gl
EK96x2nvY29ybyQdZE9/
v4jvcPhXbu9D309YWkwPYfTTD4S3fc9BtZqu3cfMLydFbu9Igsj01PYMjx0rUA9Z1qgvSmWSj3BBlm8DcuvPW
BNHr0e0YA9r5eIPMbsLL26mIi852wHPvVfOD1wWPK8gHECvSVsgr3bM7i9GNleu+Lbfj0UDY29VU0hvv3xUzx
aiAY9DtccPfk+BD0M+iy9IZABvRr35Dz0psy9G7VWPVmkod26GAE9JbptvbtnL73QTFa9+fs3veiXx7v76o09
m130040zRT3zog09YTimvAln071mTac8BPmCPcVQ8LzJ/
yY9EdYUPSJpc704YWA90CaRvRmzOr6bnjg7vLHWPBcI0ryxdKq8WsZ9PIINOD00Rz09MVeIuw==" ,
    "engine": 3,
    "alg": 0
  },
  "Hits": [
    {
      "id": "A5204509-9596-EA11-8A9A-B06EBF8119EF",
      "person_id": "2",
      "person_guid": "D5E68CD2-F4D7-EB11-AC0A-B42E999B6711",
      "sim": 0.249181,
      "timestamp": "2020-05-15T10:15:44.807",
      "surname": "Doe",
      "name": "John",
      "patronymic": "White",
      "comment": "Senior developer",
      "department": "IT department"
    }
  ]
},
"ProtocolsTotal": 1,
>Status": "OK"
}

```

## 2.6 Response parameters:

Parameter	Parameter description
id	Identifier of the captured image
camera_id	Identifier of the camera that captured the face
camera_name	Name of the camera that captured the face

Parameter	Parameter description
timestamp	Time of the face capture in UTC+0 format
age	Age of a face
gender	Gender: 0–undefined, 1–male, 2–female
emotion	Emotion: 0–undefined, 1–rage, 2–fear, 3–happiness, 4–without emotion, 5–sadness, 6–astonishment
glasses	Glasses: 0–undefined, 1–tinted glasses, 2–without glasses, 3–glasses
facialHair	Facial hair: 0–undefined, 1–beard, 2–bristle, 3 imperial beard and mustache, 4–imperial beard, 5–mustache, 6–absent
hairColor	Hair color: 0–undefined, 1–brunet, 2–blonde, 3–chestnut hair, 4–ginger, 5–gray
hairType	Type of bald head: 0–undefined, 1–bald, 2–with receding hairline, 3–without bald head
headwear	Headwear: 0–undefined, 1–cap, 2–bandana, 3–beret, 4–beanie, 5–trapper hat, 6–fur hood, 7–glasses on head, 8–hat, 9–helmet, 10–hood, 11–kepi, 12–headscarf, 13–without headwear, 14–peaky cap
liveness	Artificial face: 0–unknown, 1–no, 2–yes
evasion	Face concealment: 0–unknown, 1–face not covered, 2–mask is on, 3–tinted glasses, 4–face partially covered, 5–balaclava, 6–other mask, 7–mask isn't on properly
temperature	Face temperature in degrees Celsius
fir	Face vector data
data	Face template (vector)
engine	The used SDK: 1–Cognitec, 2–VISIONLABS, 3–VA, 4–STC

Parameter	Parameter description
alg	The used face recognition and search algorithm: 0–ALG1, 1–ALG2, 2–PRECISE
Hits	The recognized face information from the faceDB and <i>Axxon PSIM</i> databases. If the face is not recognized, the field is empty
id	The identifier of the image from the reference face database
person_id	The identifier of the corresponding user in <i>Axxon PSIM</i>
sim	Similarity level of the captured face with the image from the reference face database
surname	Last name
name	First name
patronymic	Patronymic
comment	Comment
department	Department
Status	Status of a completed request: OK–successful, if error–see <a href="#">List of possible errors of the Face recognition server</a>
person_guid	Identifier of the corresponding <i>Axxon PSIM</i> user in the <a href="#">GUID</a> format

## 3 Add/update a user in the face database



Also see [Update user in the face database](#)

### 3.1 General request format:

POST http://AxxonPSIM\_user\_login:AxxonPSIM\_user\_password@IP address:port/firserver/CreatePerson

### 3.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
objectType	No	Object type
id	No	ID of the user in <a href="#">GUID</a> format If the ID is not specified, then it is automatically generated for the newly added user and returned in the response
name	No	Name If not specified, the person is added without a name
surname	Yes	Surname
patronymic	No	Patronymic If not specified, the person is added without a patronymic
department	Yes	Department

Parameter	Required	Parameter description
comment	No	Comment If not specified, the person is added without a comment
createPersonInAxxon PSIM	No	The flag for creating a user in <i>Axxon PSIM</i> . If the parameter is equal to <b>1</b> , the user is created in <i>Axxon PSIM</i> on behalf of the <i>Axxon PSIM</i> operator If not specified, the <i>Axxon PSIM</i> user isn't created
image	Yes	JPEG image in the base64 format
fir	No	Face template (vector) If specified, the face is added to the database without any verification (face search, face vector generation)
additionalParams	No	If createPersonInAxxon PSIM = 1, you can specify the <i>Axxon PSIM</i> user parameters from <i>dbo.OBJ_PERSON</i> table (see <a href="#">Base Axxon PSIM database tables</a> ) <i>Note. Parameter text values must be enclosed in double quotes</i> For example: "additionalParams": { "auto_number" : "A123AA77", "pin" : 33, }

### 3.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/CreatePerson

```
{
  "server_id" : "1",
  "objectType" : "PERSON",
  "id" : "BC06491-C30D-450F-BF45-F9F5D680A501",
```



AIC+yPgvFP8ADnR4Ij3264Wae9tLiKKOKW00baZ7doWd9xYeWok8zBwfkHQ+3XH7RtlokD6jeeFNdSFj5c1v  
 G0cSoBxNceaI3LhdwbYYLA6B+Sa+R9J+Lun/bb9YrZtVigtciNo/  
 L+wABwcTLIQPP558vgQZwTiuI8U+K1hQ3VvewyW99GGeK4me4jt/  
 tPWLyBIvXGC29c7eRkCutSU5x5KM4zbXK9bJ+69VbVaq/zMMJjKWLxdR1pX9nL3VD3F9ndX1/03ofd2L/  
 FjSPEKz3uka7JJp1wFkgR3RZIX+Zp4ZYVmkI2kqA5ZS+D8nGD1FL4we5ggaKcGRjIqMhwV2LmsSfv7s5xgbcD  
 Tmvzc8JXx0u6ht9P0+IxXCST3MmeapYnDR77fdNkku+zEibfmgGDCfanwoKa2unwvDdJdPHezW9tJExeVbUx  
 GffnZ503zYyPkk3bm/u89EYYiTSqRXI/i0tp7vX5/wBXR7UJ4ZyXI5c3S7vrp/  
 n+P3eh674otLVYhMJPt4nPmKSQxjE3vL0zbv4yWzk/  
 3a+cvFP7QV74UnuIrOG05LYPEN16FMJG0KQv2SY87idoIzsA9Mep/  
 G1bjwzpIvrSW0K07hmUTSIN0Uw8sLciFskszkF8j0B8pGK/MrxRc3FrPcTeIJDdX6xkjW5cosyrynksRht/  
 mYYHGw7cbs1x1ea0Ip0q03azV3sn/maVakKV0dSpFyhFXklo2r20fc+o/C3xa8baxqYvb/  
 xNdvFL5j2OnWduYRCrn9+sk63LebnEAQ+Sm3a3yndgeGJ4501JIIJNWTUpJJJLhrhplhXCYLEyOpPmWMLgzuer  
 ZWAS0p+ENL+MXgjRoVsrp9YtNcs1ZDDa2rSQGT8RyJOGQz58r0AibABYQ3FS6+Icd6k0N2L60d5Jb37e8sdx  
 bRkp5eyHcCruPM3IG+by1Hytmvqlep7jSSlo21ovh1f9fmrFKY7PMHG9J0qnJKydpNSs7a3tp/Xy+q/  
 GF5EwtruxRpBYaRcWdLNI6bjGXL30wCtLtMyy24KE/  
 L5XLvuG38Qv94Zi0P4j6ibSCa0216P7cL+SQlbaI7p447cgeX9nNwu5xM3m+aDsTaN36zeG2fWNNt7vSoZT  
 p98rxR3BjLR5Nm0XKpbyPJ5KfMnlv5h8wM/A2c/  
 G37b3hb7HfFdbWLe2WytP31vSGtkImKyPDospupJ8RcuYiBAYgRuY+d8vPiSyicMfh6iqxbhUT5U/  
 iaS6J637Ld/iYfHRR0pUqNGXsZx5ZSkM3y2WvNb5eVuh8C6QgkCR7dqAkSTE52FD8uExzuyx+  
 +u3Hf0R9pfs7aaJ/FWjm6HnaZJHeSSnGJPMtbS4aLZFLjKXbcAV2nbnduyPjPw/  
 KI7mHzE3pcTQRJDnaRHNNcxGTftb/  
 AFfk527fn8wDK7SW+9v2f9J1JPHfw+i0m1mvtW1DwbnS9F0eOM+TftepDAzTXPziFU8xGQNBVL8AmG4da+jwm  
 GqyxNFVivk1czW0dt0v9fij5GLULQxWLV02jktVfblPV/8AhLNO/wCfC9/8B2/wor9Tf+GGvjF/0Lun/  
 wDgvj/+SKK+o+oYftP/AMC/r+n6GX9o4nvD/  
 wAAj5f5f1pb+YbxH42kutQldSio44o4kjdZJKlxJKz7VyXGwbdnybPvNnitpl7cXa3U3mlwtlcMqdC0hChEBz  
 1bnnbkYzgzl7ofmXcwZmVwSyApkS4+9l967NvGThs59ufcfht4Pt737Abq184SzkRRyIsJdKmjNtP+tdjBY  
 Dbs0A3b57B4enUhGpJpm30dlfTp8z1qNecqsIvls3ba39bf1pb7P/AGZ/gzqur+E/  
 DVwbaK031GSe4utSnGIIiWQRD0xYGMx5bY/zebubKIUBb7rvPgXdajZy6Po13dGeKEs1xb25e0nLL8v2Z/  
 tC+Zt2uHwFC7l6545X4CDTNF8J6do83miRLmSXTbcnNttm2+bDcwf8tli2ReUfMQjFJ8v0K/  
 Q3wbp9zFDBeSLHZM0YWIRgSW8gIwxit8J9mKGLu/eSmXcvK+X82VevPD1ocig0mnacVJP4d1p/  
 SR9dhYueGnFbyikvmo/  
 kfj7rHwE1PRjq0+ueI9e0m4CS2drBaieK09FwxWea4kjl4svLiKRmMlvtDgSjg7vSPH8JfC/  
 hi1ml8QazP4/  
 intoIXy7rizktJlFwY5Z7iU3uDmUbFEJ8wxN8ybRX60Xngv5d7bvWUa+kAn2RRWsaxymbYS0xfz8bCq7Plyd  
 z9Mccc3w+svMhTTNDsPsrPMBhgsNtIqkp507ZE3nFJS0V2H0PvVccwrSai4UFfS6pJPps76PT8WYYTKYUa/  
 0LUXtH795N6aPTTT90U+Y/  
 hT4T0Xwvq0V7DGkkkrzlyRhB0oikwYYWvmPs4LAMAPN3fctZz9BaHPqVn4k+1aYtvBcLHcRq6WyhGjvQnmKk  
 YfEwwQgMQzF9/IUqQejsvh5otldTyhjBM4V/LMRcIYtxwjecnDbyDwvQcHIPEtvs08EsBxI0rxKcZ+VSm5+G/  
 2hx9ec4rb285e6+WzsvhXL/l/Wlvdo4KjGpBrnunp7z/rp/  
 WhwHx2itdf0G0imjYy6ewe4hWQ4E5MJDN5FH7wxsduPl2feb0R8veIvCWj+MobWDWdFhvitoy2uB5ckTRLGAY  
 zKjY3fJn5SSEHXFFYPjnS2uLS7gjUSiRyPLm9J27CocoiwZbcTlGf3q4440ePMNHsII5gGEcgiXyXgGF92QWz  
 gLCMcDDZz1G0fHqz9njqE3ooyTu/  
 RL+u53V8PGVKcYpttalFqunU+M7D4Igg6LbdQ+FYPtVrLILtC3WrK0ciEqAstsdOy+wIo+aXjfyxcV67p37PNj  
 qN5bX3izw3YNGjLLHBpcwgWZlWsblliybYjs2qU040/3cc/  
 U2n+HtPjAdoLds28Fz5m052D5Q0ozzk8HARqHhMKRLHkMza4KrFtxjB38xsjrjheh45ruxGN9pQqQpySnKNo2  
 3v7u1tfl5LuFpVMkoVpXrUfd0vZuNkuXZ/1+J4Te+GdE0yykTTre00UWlq+  
 +FLV47kRBFt4oVCxCB0BkEk80yb00wBRu/  
 Mr9uSxiTwJp0rW8kd9Lp2qpfmKN+UMiJFJAGw2z0Qxk2n0wAxjgj9V/  
 FrWZSSKR4zM0oLUGHzd6xD97ECZF2hvMtNddACrY4/  
 Jz9s83F5p3hrS90wYJtZ1dNUT44NgWG5bSPsrt+8bcLXyrjaMKH84gGPHzfHQnjKWZYao5txjVT1V09Laq+/  
 m/kdzoUcHg61KjTiouFrTjyV2lpLdPb7j8xrTSrya+ha1C7ZNPijUk48qVxd3IlwfvbCpTHy5zu3DG2v3F/  
 4Jk+DNN1TxrpXiTxPaNqs+l+Cn1jQLWkCwrafq0bXiC9F15dyZGRoInCeRETkrvAGT+Ndspgv7mNYmZLV7qCM  
 K+0Mb0LY00NrEfaPtR03nyxBj955mV/cD/

gmXPawEmpPLP8AaofAH2e0McPnEvEdVluMguPLI8+FcZbcRu44Ffp0VzeIqUfaW96SvZW6J6H5hmsVhKeKxH  
 HSo4uT5vejdnT5n6z/wDC2fgv/P8A6j/4Mf8A7nor52/ti99NQ/8AAMf/AB6ivrPq1LtL7/6/  
 p+lvhf7VxfeH/gC/z/q/pb+YfXvDTxNdeWoMqWd1cRnZsErQeV+4L7j5ZkEnD/  
 Pt2n5Dk4+0vgT8MLPWt08IQ+Frd9Q1y/  
 027v760ZGItZJ4xblrNLvMqLJakuN5TMxmHyR7Pm+ZPGXmWt7dw9xIIo4b97WMDk4BEKbyfkE5LZBVthQD58  
 /L9a/sKftC+FPA9i3wr+IHIDTNE1C/  
 wBc1G40TU5rCS+1gWwSWX7PY20ZjZ44P9HW5ka9iDceBtgCEH4XL69BYfGSacvq8HKmk9XZxZuv6Z+sYelgv  
 bU9JfEvtvtHy82fZfgHRJ9GmsDqUE1peRXU8E2nT25UQmBoxmG5L4kVt2M+RGRtXAYH5fuzwvc+c1rEzDHFPP  
 8rFvMEJiC8Kp2537wTym00tfJWt+MvBNudC1C/  
 wDFWmXmt+Itk1naR3lmZpLeeZorIm1FyZLaaQRu0i0pdjhc5jJb1zWjr8waXzjtEbLFCpmQuvD7mwm4gY2fKS  
 M8HkDjgrYnDV3f2NRVPSycnyp6br5H22BhhY0nKnzOVk4tybV1y9PK3y/E+s9KmkUy7m2t5oi2F2o8QkM/  
 l7+Ebevlbc88Pu3KeAvII4CIPPVobtWnUwpJw+DGMn5R+AweDjka58y0vXTYfZbhLj95H5xJHR/  
 M8vGefl2bSP4s7vbl9/4sjELuXc3bsxhccgZxvJABxjKfXnpznKn8cfU7I+9JRdrPt2/  
 yX9W0k8ReIdK0q0R5GWW5x5Swb9jKXwMmTD7j9EHpx25LQobi61CJ1c3YMc0qWca5um8zacW8QJ8/btG8/  
 IV+Tj5uPPp7051W6893lvWnbzI1hTef3XLqW8z5c71AJ54PAwK4PxF8TvG3hDVYoPDXgHxDqL380L3NjmJ7BU  
 2hZPtRtpPs4m30duxvME0dy7MnuW69V+Z0KlCLTV7rzPetWsp5lla9iubeC1ac3VvLgYXDxkJt82MH90yBG8p  
 MyGQsfubBu+evE922k3FtqmnQXKaPdSXCQSS7klke3aDzfMt8EwKhnTy2LOJgzckU5ta9+0f4i0Xsof+Eo0j7  
 XcrHLJHZPLFDfzykR7xf3kdvcFhbHy9im2G/  
 wA2T7nIr5fX4hfFDx14rN5NBRJoWrn5LWIF4rGK1PCKnkxDdIs/  
 zuFteYwSvy8eRjEpYimns2k+nSjvTV6kE9nJH2voerG/  
 0+1uVkeGf0V3Dcn3eD3yc8ZAxjvXSzX8cVuzSMFYIdoLZyMHdzj0ACD0PuK8A0aa7srcIrtFggMvJAZcg4GQB  
 zngZ6fnoajrcxiUPchCrJk5+7kL2y0Dj88Z46VTowh0Mle6atdtrotV8v60tnXk1WhR05J6S01tZPfp8uxn+M  
 dUIjvbtLgw/ZoLmYN5fnAIoTd/  
 FHg4PX24HGa+CPijPZatG91qMsLWMtv000rXarbWcE0oUM1wWmnlGMogj05vNy3+r2Yb6I8deLyLS9tLSV3d4  
 ZEm2R+dvDzW8IjKjG3cJH0cn0zaAeo/Kr46/  
 EgXPxF8Q+FidQuLzwl4d05NMu9KikVZ9QvIVkkumhTzDILHyhv3Db5wJlG8AcFV3xlnWvNu6Lyj/  
 meHmlT2Mjpw50XW//AG71+eh5XYyRTalqEUirmPvPvjngYXNjerG0nlbtXQEYkF3n96PLP2fy1/1wf5f18/  
 YBu00XV/  
 B8t7cSW174zL8YaA2l25Ex0uG10sX9tdS3GV+1ecIZIkG8i1CbmcTPs2H8efA1kpmjBRvskUkEttbhtsM6hpi  
 skIOTbtgtjzUPnEkRgkFa/XL9lrTNKtPFxwkSaWXSbq/  
 8AEWryNqrX0+KJZ9JmsLjaAxxhRH9tL7vOXftKbV+8v0eAxVWhWoqDjZTVrxv/  
 AFsfa5LDC4jCV1JSu6btadv+fZ+meJv+el5/35/+2UV1X/CvZf8AodrP/vtf/kiivrfr+I7x/wDAf6/  
 p+h+ffUMP2n/4F/  
 X9P0P5QPirRb3Ws+I0W5aXzvE2oX0UhXYfTibua0zRRubIKo7bgQB02nApvwM+IUfw1+NPw+  
 +Js3h+Hxld+HNT1B5fCciXs3i6HUIIbc6LEz2d+9k7bA5vYr09MZ2g2jbgY8H4ovDe01yJPMG6e2DmMxNIsv1  
 NIJmTc2zeLjaFLP8AcJ3HOAz4U+OpPhT8XPhj8XYdEh8V3Pw58Rf25H4WvLlbKy16P915tjc3bWmopbAmKmp0  
 2nXuzLDyDuzXz+SYXBfV6lSKm6lWDbUqjlfV3X8L7bn1MsXLCyXtG1NwtFTX3d0/8/8AM9j8aX8mqfEPV/  
 ibZeGhofim38Ww+Jd08I6je3tqLa01mvdRh8HPLIFWNYbdrpl1IabHgRY0l/OPL/  
 bP4X+LLLxP4A8E+PrK4sCnjzSk8QtHYhliTq5FubEXDSv9sazuYbmEzrDaFvvNbpuAH4V+PviprvxX+I3jf4L  
 a9p9lpN9468XXviG30Kz2vD4at7qCa3td0tpkhtVv1sre4u42uBaaeJ/  
 PU+RDsw335+wB46t9V8Eal8JtaFzG3w1g1HU/  
 BbTWxto59IvLxprpTpZTJJsFvMsUsNwTJ5huHRYUAzyUYwrvE05whyw0jyxUX9nqtep9Hluc1XGKU4811Zcq  
 9L/X4bH6sabrKyxAMHfAUZYlQM5yVyrBs4Hpt00ua3Y74R0sgWB0KSRvDLKI2feFctG/lv/  
 AKvDZULyWHOAK4CyuFuFCRyGwQ+Wq0E2MzPHG8i43chCyKrlsPy2F7bvn/ZPKuSsUwhDk+YpYKcLjgE5B2/  
 XKjrXl1oRpV6cYXs2t9e3+Z9jgcVVqpSm4393aNT0r/mer+GbWzgs0MKyLerI0onjm01fNb03y/  
 KIIG3rvXjPHIrtUN3Ess1wQ8UsEqzSm2eUMcfut5Rk2feYrndu04jG018myeP/  
 AI1XM93F4J8JaTqFrbGNXmuNQisDiTf5BRZbKbfny5d2HG3C5+8K4vxH4n/  
 ad1a0Wz1S3/4RuAscX6DqdpM14uRvW5c28Xl+SNvkEK2fNl90d+SpU9yk1GpJ2g2rpPzXU9NzrzXJScVUlPbt  
 JpPzue7N8GtA1/  
 VT4j1fSDd2tsb5p5Lk5aKmm4MRiN3HNv8A+eLmBV6fvQcZwsqw0C1tG+zadp9s1layXSxta28Ub2izbNouH81  
 vML+X+7I2EeXJ3NfK974Y+MF88Nsde8W21vqRDT7vEKiKL70Vz+5htwJ/  
 OE7ZDNH5YTgSFyUNR8GeKYcQ3XinXbFrRQIfsWot/  
 pjPjzWuwMCQx7E8sHG3zZ0ea5q2WYxP6xWqQaprmajFLS0ezf8AXyv6GfWw0jT9vWq0pKn78LGCv9tFZ97/

APDHuGu3Nrp0/  
 wBkMkId2mbe8qxk8x5AjAcHbnJw3fuTXmt7d3M7ypDGgJQHexl2DDZC7fkOc7W6Y6cjmUtpCLKYbnVdavn6S  
 Ikxefu3jOPM3uZ23ltsfJUY2nrnIv6ld29rGiGRmysm1BlQY0C+Y8knzBdoK7QV07ceQ0vN0oqUHUltDVnkY/  
 EYVYoyglzRV17t+kenU4C5vbXTtcSW4e0hsoRPqmsPcMJBdpm17JrmdRswRF5qqylk3eYh3HZixv2j+LUmi/  
 ET4neJtI0/Sdbh8W+JvGY06710xWWOLS9WXTTrfT7i1VvM3G0FrcsOU8zzVwYvLy/  
 6IfTKfFi3+GPw58QDTYbRtS8faPq2gQ56hFvLitGW1S7nsZDn0RexGZV8vdiH940a/  
 HDRfMS3sLct5wgt4Yw4Xy1PzSOSIwX2E7yD8zZ2jkly0KX1mtCuk3BSUm1stv8AgfPTU+WzbM4zw9eFWUXWLT  
 ahb3dfdfp57pdbbo+jvAizI9mHkM0rCSWSQL5cZkd97G0HL+Wh3YCB2Aw0ew/  
 Tn9mHWZL3xt4T0TU2t105U1K7DS0JZ0mtYYUha1TKFSPtDGQq+SFTqTX5veALViYZSCdkOcY74PGc8Zx1I/  
 Dmv0r/AGR9PL/F74X3hNqhuo/  
 FNtLb3iCRYiINMkicAvGJi3lsuwqmg7m5WvfwCDxFFa61I9T4jnLP3ZNtNNP7L/8ij9H82P/Qbl/  
 wDAGf8A+S6K7T+2bn/oG6R/34X/ABor6j2MP733/wDA/q/pbzfyw/vff/wP6v6W/  
 kh1qK6eWRboAtiGAYFV2Iz/  
 ALwsRHn5M5+b5z05xxnjQ0kbbk+V4Dwxwc+Z144xt2jjJyDntXq+rwNIZncb3B0WxguuBtXGcDaBnqQc9BXmF  
 78hl03GGHF73PX80nevjcvqTw06d0k3yXtaWun9JHvNWHp1HKtLm54K6tJpXtDddTdt7h5YwWGZHhLXcOGUgx  
 gSKecbQxz1zn7wxX3F+wjPJH8f/EHgeS4mLs/GXw0mktWuZizPq+juZLe1s/  
 3WYBdi+mNw4dsC3i0yTAX8IaXdAKGMW791Lft3d5tgDfdxLPLzjGW3csMc/Xf7JWu3+h/tT/  
 s8X1ibZV1DxTLo+rLcxiQ3GmzaVeCWNXJXy/mKsSVf7in0Ye3hoRVZpX/f00/w27bIMpvF0pJu91u32X+fu/  
 bW0lm0u6/  
 s3VXgju9sMzfZzHcQRWmgigXzAiB3BhLEhAGCMYBGa6K5uVZVjWtZiz17buhHGTtxnvn0e0c1btPDs0pwW1vF  
 bRRBkvYYLwyB3eVNV1JLllchMpcCREij3YhEDESSCTCef3UL14e1qTR9djs3Tc0UrBnimTGcokDnKlvm+Xc  
 v3s14141MZU9r8NKWltLJcu9t/1PtclmNej0Ebx5Lq94q9LzB/I9DsLwIF8yFDFb8RqTIN2/  
 r80bR42hbJcYIC4wd2Rqs32lp3t7bcGUuy9ofykjySxJnGeDnGAFpVwvMttqNpPKs09I9mdjBmx8+0hdq  
 k78HgfdAAOc89hp3g+w1NdPuJLuSL7f9pjFvbgNKEj8nzVuMyLTLiRRGSDyr4zjFdNSpQjCTpyaqJXi+br7vb  
 1f9NH1mFxcK0L83vaWa01uunfTtY+c77XrqKKSyiu3ima4igliWVPNmi8pWiIZ5tu0yLTYHoTwTjGBBdX1xe  
 i08a5t74K73VnPG0ux0sJWUmPcJfnDDy12BV5b0R9ZX+ieHfDqDtD0synltJIGu2W42TnafNjG1QuSCXX0SQu  
 LDaDXC3g0vzEudQhtrZU55VtEmAW0WVJhhd+B0E+ZgLEt5vmnLCuG8+piKzpyTm2rbN6dP8v60t6P1zEUQuo  
 xn7ttnr209NEeBzLqE7PtBOWZY0PGNnU546kg4wMY5JPI8+8R3k8cM1p5ypJ0j2bnaCI7ibElvG8hdfJhkEFx  
 5kxUiPamVYnj1nxdrtpoS3kCRKJo5Z0EwL+SLGzGz92fMLL+RuXaFHXD8vx8D8UdRvW8N+NtSu7m6ijm8K699L  
 eC4MDPMI4AJUfaxSWDcvLMAASPnGM15deTq0Z05/  
 DNWdtHa62fQ8fE1ZTTqS5SRStppuuh+bP7SvxVvfjN8SLj7Dqkp8CfD8SeF9IsGs/L3eIbcxR+KyJ/  
 0Xz4N1rpBguPJkyhmwEkkt5t4csv0mC8hTsZcrnaE3cAFhn0B457c4ri9FtrhtJtHneWaYyyyyFTmUSSsVkl  
 Xcd9w2D9ouN26f5MquwA+veFbU4SepwJH97Pv6+v5jg162V040sPyRvaUb0+r2js3rc/  
 P8AMqcatdVJ0XNzJ2Um19np8z60+H1rKbVDHD5mUDtzjaLFPHQ5Eu/  
 r1XZzv3fL+jH70nh+Zvin80L5oJha2y3dxJPE5RLAXMMUbK67Sjv027hl4tohi58zK/  
 Fnw80Mw6fu+0bVaEhj5AZ1WTJLLH5w8zy9vzAFQdw+7X6u/speHhDYavr95uuLKW0qx0e4NttkkkjGp/  
 bpNu+Qw+VvtPk3N5hc/Mu3Ld0S1Z4nMVSqW5YVUo8qs9+rPKxn+z4atWp6TpwvG+qvdLVddz7C/4RbS/+g+f/  
 AABP/wAnUU3Fp6z/APfo/wCNf0X1a12L9/9f0/S3x/9q4vvd/wBf5/1f0t/LDqNi3lPuXB5x8v0zj5vb/  
 GvH9ZsSfN2nAz/AHSe/b5v519Bat/q/wAH/wDQVrxrWPuy/  
 wDAv51+U4etP21PVfEuh9tjYqrRq0fZbadYr9Dkd0jcFISPvBju/wBwjbb759c19M/  
 AG3YfHT4GajudTZeJr5sjKxpmxEE+SbkJtLhLBQLtpXI5r5103/j4i+kv/stfU/wP/wCRx8E/9hr/  
 ANoXFe5Tr1YTjJSV019LeS/z+/yKymhTvSWtubv2UT9/  
 fA8i3eh6LHLG6LaLeW8+2feL1or+eW3vEmCL5Wf0cGLZKMAHz0cDoPFukW/  
 iSyW21LypLm13f2fd+WFhJfZ5kb4bM5by4drLkC7DLWDHHLFDH/kTPDP/Xvd/wDpQa7HWP8AWQ/V/  
 wD0BK8TF1Z060uW37347rvy7dj6b6tS/vf+BP8Ar/  
 hz5x1bSvEWh6tZahps9yx01blIrGCUw294JRBk3AAkVvK8oeWcj48x+Rk1rwfFua0W2g8Q2l3YXKL00Uum037  
 jb5fm+eohUSbz5ew7kC4k4bcSvbXv/H2f+B/0rxLxJ/yEr3/rLj/WuZVJLr/Wn+X9aW78E/  
 Y1acYX5eZaN3vp/  
 wAA607+PPHk0K8xDq2o3Xyo8szsGQpv2lFML43Z075hja0TXjetfFKLXniTT7q6hijulnuIJYmiF92f3KRTbs  
 /NmXc3L/JheDnA8zf/AI+bv/ro/wD0qA/4+h9R/Om6s2mnaz8j2q9eao1GrXUdNPT/  
 AIP3+R3V7f3mquwGaS1cyzRox30r0I9yvLgF921cHYoXBwDuIHZD+0zrD6N8HvEetY2paQS6R9h37CE1qa3g  
 E5uNjH90LbeIjbnzSceZGBk/UFt/qU+g/kK+Pf2yv8AKiXjP/r98P8A/pZJWdr2T2bX5nhYjGV/Y1NY/D/  
 Ku6Pzlt7JLS1ggVhJHbC0yWUJs80xQwzNKU3PsLfaQmzc+BHned2B6n4ThejRAL90BsdckZ+nPHUn068585b/

```

AFQ/6/H/APSLTq9R8E/6wf8AXL/2Y17uJisFh6EqGjmlzcz5ui27Hzs37aSc+620tt/
kj7N+HRsayS48poJLRVEbKpufM3n54jb/
ALrf5uwYbedhXlXzx+137PsVrD4S0iwsrdJH1G1huQkqiBYJwrnZHUh9hbevJILEYxxX4yfb7/X6Z/
19xf8AobV+0HwS+7H/ALi/+gy1nkEuXN8KraVq8VU13um902p5+bUoxy3GSV01SbXXW6PoX+yL7/nws/
8AwIT/AON0VYor9c9lT/lf/gT/AK7/ANLX8u5p/wA34Ly/4P3+R//Z",
  "additionalParams":
  {
    "card" : 9876,
    "facility_code" : 12345,
    "auto_number" : "A123AA77",
    "expired" : 3,
    "phone" : "0123456789",
    "pin" : 33,
    "is_locked" : 1
  }
}

```

### 3.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

### 3.5 Response example:

```
{
  "Status": "SUCCESS",
  "Result": "7BC06491-C30D-450F-BF45-F9F5D680A501",
  "Description": ""
}
```

### 3.6 Response parameters:

Parameter	Parameter description
Status	Status of the completed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>
Result	ID of the added user in <a href="#">GUID</a> format
Description	Request description

## 4 Update user in the face database

### 4.1 General request format:

POST http://AxxonPSIM\_user\_login:AxxonPSIM\_user\_password@IP address:port/firserver/UpdatePerson

### 4.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
id	Yes	ID of the user in <a href="#">GUID</a> format
name	No	Name If not specified, the name isn't updated
surname	No	Surname If not specified, the surname isn't updated
patronymic	No	Patronymic If not specified, the patronymic isn't updated
department	No	Department If not specified, the department isn't updated
comment	No	Comment If not specified, the comment isn't updated

Parameter	Required	Parameter description
additionalParams	No	<p>Additional parameters of the Axxon PSIM user from the dbo.OBJ_PERSON table (see <a href="#">Base Axxon PSIM database tables</a>)</p> <p><i>Note. Text parameter values must be enclosed in double quotes.</i></p> <p>Example:</p> <pre>"additionalParams": { "visit_document" : "F183825", "phone" : "0123456789" }</pre> <p>If not specified, the additional parameters aren't updated.</p>

### 4.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/UpdatePerson

```
{
  "server_id" : "1",
  "id" : "FB94569E-86A6-EB11-BE5B-B06EBF811A3F",
  "name" : "John",
  "surname" : "Kramer",
  "patronymic" : "Kramer",
  "department" : "Department 2",
  "comment" : "Some comment",
  "additionalParams":
  {
    "visit_document": "F183825",
    "phone": "0123456789"
  }
}
```

### 4.4 Response headlines:

Headline	Headline description
Allow	Returns the HTTP method used in request

Headline	Headline description
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

#### 4.5 Response example:

```
{
  "Status": "SUCCESS",
  "Result": "FB94569E-86A6-EB11-BE5B-B06EBF811A3F",
  "Description": ""
}
```

#### 4.6 Response parameters:

Parameter	Parameter description
Status	Status of the completed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>
Result	ID of the updated user in <a href="#">GUID</a> format
Description	Request description

## 5 Get information about users from a specified department or the entire face database

### 5.1 General request format:

POST `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/ReadPersons`

### 5.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
id	No	ID of the <i>Axxon PSIM</i> user If not specified, the response returns information about all users
page	No	Number of pages with results If not specified, then page = 1
pageSize	No	Number of results per page If not specified, then pageSize = 1, that is, there is one result on one page
departmentFilter	No	ID of the department

### 5.3 Request example:

POST `http://1:1@127.0.0.1:10000/firserver/ReadPersons`

Get information about users from the entire face database

```
{
  "server_id": "1",
  "objectType": "PERSON",
```

```

    "id": [
    ],
    "page": 1,
    "pageSize": 2
  }

```

Get information about users from a specified department

```

{
  "server_id": "1",
  "departmentFilter": "1"
  "objectType": "PERSON",
  "id": [
  ],
  "page": 1,
  "pageSize": 2
}

```

## 5.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 5.5 Response example:

Click to expand...

```
"PersonList": [
  {
    "Id": "1CEE737C-B5D4-EB11-AC0A-B42E999B6711",
    "Name": "",
    "Surname": "1CEE737C-B5D4-EB11-AC0A-B42E999B6711",
    "Patronymic": "",
    "Department": "3",
    "Comment": "",
    "Timestamp": "2021-06-24T06:29:16.199",
    "ImageId": "1EEE737C-B5D4-EB11-AC0A-B42E999B6711",
    "AutoCreated": true,
    "LastRecognitionTimestamp": "2021-06-24T06:29:28.270",
    "RecognitionCount": 1,
  }
],
"Total": 220,
"Response": {
  "Status": "OK"
}
```

The response returns users in the order of when they were added to the database (most recently added first).

Using the identifier from the **PersonList.ImageId** parameter, you can get the user's photo with a separate query. See [Get an image by ID](#).

## 5.6 Response parameters:

Parameter	Parameter description
PersonList	List of users with information from the face database
id	User identifier from the face database
name	Name
surname	Surname
patronymic	Patronymic

Parameter	Parameter description
department	Department
comment	Comment
timestamp	Time of adding a face to the database of reference faces in UTC+0 format
ImageId	Identifier of the reference face image from the face database
AutoCreated	<b>true</b> if the unrecognized face is added automatically
LastRecognitionTimestamp	Time of the last access in UTC+0 format
RecognitionCount	Number of accesses
Total	Total users in the face database
Source	ID of the server in <a href="#">GUID</a> format
Response	Response
Status	Status of the completed request: OK—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>

## 6 Delete users from the face database

### 6.1 General request format:

POST http://AxxonPSIM\_user\_login:AxxonPSIM\_user\_password@IP address:port/firserver/DeletePersons

### 6.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
id	Yes	ID of the reference face from the face database in <a href="#">GUID</a> format * all *—to delete all faces from the reference faces database

### 6.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/DeletePersons

```
{
  "server_id": "1",
  "id": [
    "4c6ecb1d-2a37-e711-9e4b-9c5c8e763a8f"
  ]
}
```

```
{
  "server_id": "2",
  "id": [
    "*all*"
  ]
}
```

## 6.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 6.5 Response example:

```
{
  "Description": "",
  "Result": "4c6ecb1d-2a37-e711-9e4b-9c5c8e763a8f",
  "Status": "SUCCESS"
}
```

## 6.6 Response parameters:

Parameter	Parameter description
Description	Request description

Parameter	Parameter description
Result	The value of the id parameter in the request
Status	Status of the completed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>



Cdh5Jry4maIq0caRqxL3eVv3EuAgTJYHBRi0etaFt45bxBdW0kKXitriQ/  
aZvMJYCMp5e35F+9ufPzc4r5fg1a4LLxdBvLbiQepPG0AYwAM840cYgK9L8LS7Z4pZJki8qWBN23DfvfMyw+Y  
ZC+WMr79RxWFWrVd0Shbma00vrf+v0Pdy7B1as6cmm3ddLJ7d07d/  
L8bfW0s63eNZW+hWd5cfZY4GWkyg4a4UqgneWfLEeXhCgMbgH3+YY5+rP2a/g/  
DqE2ka3NZCSxt7e+umupZAZYP4/  
JZYwSAVDMTv679oBAxXy58IdHvVGGpuLnDI7tcca2s7W5mWwGMyfa5Fy6bMK005QWB404Y5/  
Vz4WeC30i1g0nTr14bAW7reu0HL7ZpQCwWPzznLXydwfyfyr08BSo+y561/rEVzQs2lZafZW/  
XyPtsL70hScZr964+7/iVlt5a7372vt3ltp4jii4BuGSN/  
JKHyoljkwFCxASBCoQb3BPmYX5V2jMEUuqPHDDGW92LFj8u4rgbdv5888d+nb3tjBaW6yWYhkhiiWBybjktB  
nMgUqcb95+TPy7Rljzjk476Fpt1s372M/PGV2hs9MSZ5/i/g4/KuupJ+zktNu3p/l/  
Wlt4VZuStTzVsLdS3ds0a7fLX5sg/xgbR2AxjJI5I59gTizBGL+0bgGGQVxnJxwM5HTp0/  
Lvraj9kqiSzL553houP3R+XHzH72/  
knhcBc8544+4u0jniCOZf0EHuIPIUR7Dg4JwG3jHAzG4yBiuB02pvNtRbW6Rl+I53a2e3EaqbkhhKV8zyxD6L  
tXdv8AMH05SNv8QrxjYi+WK00QNdrDc+ViIKrLFu3pv5bIXYc/Lzj06vo/  
7NDfackn2fzPXWcpCSAy+WEx8x6L9wH3Btx3J48Y1gWqyWzKpSly961hJ5hU/  
ZVuIpX+0DBXz8G2w0f7r7wPmcYPn4+hehWq00/  
brJzQvqua8fs6X0W1zlddpXm1yrWXTT1Pzn+0HhK31bRrm1u7ZI760sNdvoHLYSyyrbCw89S5WmgfNGR97YeM  
Nkmvxw8ao+mXYhhVRDJF58dwJD+8aQkSRGMKQBbtUBzIFM837sYXn9vvi9a3d9GByRm41HU9M8RQaXLG+wSWO  
z2cd7MyhXwIxDbFV3YfzG+ZdmW/D74mqLprmpaeki/  
8AEuRLNohJvMc8Ty+bIemwynGEI0NhyzckeZgK16tKFa3tJPS2ivZPZersuqPAx0HW5qkP4aTLJ76WS30tdrt  
/kqXhJxJeadqFtJ57yW0sU0VyquYyImmfy5G8/v05AGAnQ54/Rv8AZ7+K0sx+M/  
B+oJqV3aXDXNrdhPplqhEwnxby/  
8AaDG5g+0tbjyxbIdnmGWf5o8Hd+UGkyzpcgnLx9GTpuzwCGwSu3B7c8ZIXmvr74QXFzb6voEyyCSysbk3bWm  
HSR7hY3mtkwdJAYwWikv/3L7tw0AEIb3sFhKWgXUK9HmjU50/i0v6WPj/AHsbTxmGqax5XFK0j+y9/wCvPQ/  
ob/4aBvv+gprn/ggT/wCW9FfmH/w0H4s/6E03/wDBxN/8q6K+n+t1/wCb8PT/  
ACR4P9g0NPdqdpTv+75ef9aW+K/E0vXd/  
qTzSyqbeEuba3XChVaRo2JbP8ZgVh8ox0yQMnLZNSaSYn0EzWu4kDsee0c/r7Vl6ncPFcS/  
eeSMG1dmJjVxDLNItwoIfAm+0LSgyE8sESPn4cF35k5EHMKgE5BLnaBg8fKc47c8e5r5eVFWfInzaW18z3sP  
hqn1iLKfwqSbSVltHt6nrGjyfaGYgquzywzX+7b8Y4/2WJ9zz7+8eB/  
Dv8AaWpWRuImLQuPjjc7ZCwILNfw/  
Lt45yQMcGvmf5CdWH+gtJcBpYkDoCqghnB5Bb80eME9TX6GfCvT7XwtpunX2r2bS11Vo5FkDsnkjL7wEGzPm  
IVySWKnPQVFKLwPzh0UoAMWm1bf4f8/16o/QMt5KUVVsvcSeuy0j6fL/  
AIC+yPgvFP8ADnR4IjP3264Wae9tLiKKOKW00baZ7doWd9xYeWok8zBwfkHQ+3XH7RtlokD6jeeFNdSFj5c1v  
G0cSoBxNceaI3LhdwbYYLA6B+Sa+R9J+Lun/bb9YrZtVigtciNo/  
L+wABwcTLIQPP558vgQzWtIuI8U+K1hQ3VvewyW99GGeK4me4jt/  
tPWLyBIvXGC29c7eRkCutSU5x5KM4zbXK9bJ+69VbVaq/zMMJjKWLxdR1pX9nL3VD3F9ndX1/03ofd2L/  
FjSPEKz3uka7JJp1wFkgR3RZIX+Zp4ZYVmkI2kqA5ZS+D8nGD1FL4we5ggaKcGRjIqMhwV2LmsSfv7s5xgbcd  
Tmvzc8JXx0u6ht9P0+IxXCST3MmeapYnDR77fdNkku+zEibfmGGDcfanwoKa2unwvDdJdPHezW9tJExeVbUx  
GffnZ503zYyPkk3bm/u89EYYiTSqRXI/i0tp7vX5/wBXR7UJ4ZyXI5c3S7vvp/  
n+P3eh674otLVYhMJPt4nPmKSQxjEf3vL0zbv4yWzk/  
3a+cvFP7QV74UnuIr0G05LYPEN16FMJG0KQv2SY87idoIzsA9Mep/  
G1bjwzpIvrSW0K07hmUTSIN0Uw8sLCiFskszkF8j0B8pGK/MrxRc3FrPcTeIJDxBd6xkjW5cosyrynksRht/  
mYYHGw7cbs1x1ea0Ip0qe03azV3sn/maVakKV0dSpFyhFXklo2r20fc+o/C3xa8baxqYvb/  
xNdvFL5j20nWduYRCrn9+sk63LebnEAQ+Sm3a3yndgegJ4501JIIJNWTUpJJJLhrhpLhXCYLEyOpPmWMLgzuer  
ZwAS0p+ENL+MXgjRoVsrp9YtNcs1ZDDa2rSQGT8RyJOGQz58r0AibABYQ3FS6+Icd6k0N2L60d5Jb37e8sdx  
bRkp5eyHcCruPM3IG+by1Hymtvqlep7jSSlo21ovh1f9fmrFKY7PMHG9J0qnJKydpNSs7a3tp/Xy+q/  
GF5EwtruxRpBYaRcWdLNI6bjGXL30wCtLtMyy24KE/  
L5XLvuG38QvJ94Zi0P4j6ibSCa0216P7cL+SQLbqaI7p447cgeX9nNwu5xM3m+aDsTaN36zeG2fWNNt7vSoZT  
p98rxR3BjLR5Nm0XKPbyPJ5KfMnlv5h8wM/A2c/  
G37b3hb7HffDbWLe2WytP31vSGtkImKyPDospupJ8RcuYiBAYgRuY+d8vPiSyicMfh6iqxbhUT5U/  
iaS6J637Ld/iYfHRR0pUqNGXsZx5ZSkM3y2WvNb5eVuh8C6Qgkcr7dqAkSTE52FD8uExzuyx+  
+u3Hf0R9pfs7aaJ/FWjm6HnaZJHeSSnGJPMtbS4aLZFljKSXbcAV2nbnduyPjPw/  
KI7mHzE3pcTQrJDnaRHNNcxGTftb/

AFfk527fn8wDK7SW+9v2f9J1JPHfw+i0m1mvtW1DWbnS9F0eOM+TftepDAzTXPziFU8xGQNBVLv8AmG4da+jwm  
GqyxNFVIVkc1zW0dt0v9fij5GLULQxWLV02jktVfblPV/8AhLNO/wCfc9/8B2/wor9Tf+GGvjF/0Lun/  
wDgvj/+SKK+o+oYftP/AMC/r+n6GX9o4nvD/  
wAAj5f5f1pb+YbxH42kutQldSIO44o4kjDZJKLxJKz7VyXGwbdnybPvNnitpl7cXa3U3mlwtlcMqdC0hChEBz  
1bnnbkYzgzl7ofmXcwZmVwSyApkS4+9l967NvGThs59ufcft4Pt737Abq184SzKzRRyIsJDKmJntP+tdjBY  
Dbs0A3b57B4enUhGpJpm30dlfTp8z1qNecqsIvls3ba39bf1pb7P/AGZ/gzqur+E/  
DVwbaK031GSe4utSnGIIiWQRd0xYGMx5bY/zebubKIUBb7rvPgXdajZy6Po13dGeKEs1xb25e0nLL8v2Z/  
tC+Zt2uHwFC7l6545X4CDTNF8J6do83miRLmSXTbcnNttm2+bDcwf8tli2ReUfMQjFJ8vOK/  
Q3wbp9zFDBeSLHZM0YWIRgSW8gIwxit8J9mKgLu/eSmXcvK+X82VevPD1ocig0mnacVJP4d1p/  
SR9dhYueGnFbyikvmo/  
kfj7rHwE1PRjq0+ueI9e0m4CS2drBaieK09FwxWea4kjl4svLiKRmMlvtDgSjg7vSPH8JfC/  
hi1ml8QazP4/  
intoIxY7rizktJlFwY5Z7iU3uDmUbFEJ8wXN8ybRX60Xngv5d7bvWUa+kAn2RRWsaxymbYS0xfz8bCq7Plyd  
z9Mccc3w+svMhTTNDsPsrPMbhgsNtIqkp507ZE3nFJSJ0V2H0PvVccwrSai4UFfS6pJPps76PT8WYYTKYUa/  
0LUXtH795N6aPTTT90U+Y/  
hT4T0Xwvq0V7DGkkrzLrRhB0oikwYYWmVwPs4LAMAPN3fcTZz9BaHPqVn4k+1aYtvBcLHcRq6WyhGjvQnmKk  
YfEWwQgMQzF9/IUqQejsvh5otldTyhjBM4V/LMRcIYtxwjecnDbyDwvQcHIPEtvs08EsBxI0rxKcZ+VSm5+G/  
2hx9ec4rb285e6+WzsvhXl/l/Wlvdo4KjGpBrnunp7z/rp/  
WhwHx2itdf0G0imjYy6ewe4hWQ4E5MJDN5FH7wxsduPl2feb0R8veIvCwj+MobWDWdFhvitoy2uB5ckTRLGAY  
zKjY3fJn5SSEHXFFYPjnS2uLS7gjUSiRyPLm9J27CocoiwZbcTlqf3q4440ePMNHsII5gGEcgiXyXggF92QWz  
glCMcDDZz1G0fHqz9njqE3ooyTu/  
RL+u53V8PGVKcYpttalFqunU+M7D4Igl6BdQ+FYPtVrLILt3WrK0ciEqAstd0y+wIo+aXjfxycV67p37PNj  
qN5bX3izw3YNGjLLHBpcwgWZlWsblliybYjs2qU040/3cc/  
U2n+HtPjAdoLds28Fz5m052D5Q0ozzk8HARqHhMKRLHkma4KrFtxj3b38xsjrjheh45ruxGN9pQqQpySnKNo2  
3v7u1tfl5LuFpVMkoVpXrUfd0vZuNkuXZ/1+J4Te+GdE0yykTTre00UWlq+  
+FLV47kRBFt4oVCxCB0BkEkg80yb00wbRu/  
Mr9uSxiTwJp0rW8kd9Lp2qpfmKN+UMiJFJAGw2z0Qxk2n0wAxjg9V/  
FrWZSSKR4zM0oLUGHzd6xD97ECZF2hvMTnDdACrY4/  
Jz9s83F5p3hrS90wYJtZ1dNUt44NgWG5bSPsrt+8bcLXyrjaMKH84gGPHzfHqNjKWZYao5txjVT1V09Laq+/  
m/kdzoUcHg61KjTiouFrTjyV2lpLdPb7j8xrTSrya+ha1C7ZNPijUk48qVxd3I1wfvbCpTHy5zu3DG2v3F/  
4Jk+DNN1TxrPxiTxPaNqs+l+Cn1jQLWkcWrafq0bXiC9F15dyZGRoInCeRETkrvAGT+Ndspgv7mNYmZLV7qCM  
K+0Mb0Ly00NrEfaPtR03nyxBj955mV/cD/  
gmXPawEmpPLP8AaofAH2e0McPnEvEdVluMguPLI8+FzcBcRu44Ffp0VzeIqUfaW96SvZW6J6H5hmsVhKeKx  
HSo4uT5vejdnT5n6z/wDC2fGv/P8A6j/4Mf8A7nor52/ti99NQ/8AAMf/AB6ivrPq1LTL7/6/  
p+lvhf7VxFeH/gC/z/q/pb+YfXvDTxNdeWoMqWd1cRnZsErQeV+4L7j5ZkEnD/  
Pt2n5Dk4+0vgT8MLPWt08IQ+Frd9Q1y/  
027v760ZGitzJ4xblrNLvMqLJakuN5TMxmHyR7Pm+ZPGXmWt7dW9xIIo4b97WMDk4BEKbyfKE5LZBVthQD58  
/L9a/sKftC+FPA9i3wr+IHIDTNE1C/  
wBc1G40TU5rCS+1gWwSWX7PY20ZjZ44P9HW5ka9iDcEbtgCEH4XL69BYFGSacvq8HKmk9XZxXzuv6Z+sYelgv  
bU9JfEvtvtHy82fZfgHRJ9GmsDqUE1peRXU8E2nT25UQmBoxmG5L4kVt2M+RGRtXAYH5fuzwvc+c1rEzDHFPP  
8rFvMEJiC8Kp2537wTym00tfJWt+MvBNudC1C/  
wDFWmXmt+Itk1naR3lmZpLeeZorIm1FyZLaaQRu0i0pdjhc5jJb1zWjr8waXzjtEbLFCpmQuvD7mwm4gY2fKS  
M8HkdjgrYnDV3f2NRVPsycnyp6br5H22BhhY0nKnzOVk4tybV1y9PK3y/E+s9KmkUy7m2t5oi2F2o8QkM/  
l7+Ebevlbc88Pu3KeAvII4CIPPVobtwUwpJw+DGMn5R+AweDjka58y0vXTYfZbhLj95H5xJHR/  
M8vGefl2bSP4s7vbl9/4sjELuXc3bsxhccgzxvJABxjKfXnpznKn8cfU7I+9JRdrPT2/  
yX9W0k8ReIdK0q0R5GW5x5Swb9jKXwMmTD7j9EHpx25LQobi61CJ1c3YMc0qWca5um8zacW8QJ8/btG8/  
IV+Tj5uPPp7051W6893lvWnbzI1hTef3XLqW8z5c71AJ54PAwK4PxF8TvG3hDVYoPDXgHxDqL380L3NjmJ7BU  
2hZPtRtpPs4m30duxvME0dy7MnuW69V+Z0KlCLTV7rzPetWsp5lla9iubeC1ac3VvLGYDXdkJt82MH90yBG8p  
MyGQsfubBu+evE922k3FtqnmQXKaPdSXCQSS7klke3aDzfmT8EwKhnty2L0JgzckU5ta9+0f4i0Xsof+Eo0j7  
XcrHLJHZPLFDfzykR7xf3kdvcFhbHy9im2G/  
wA2T7nIr5fX4hfFDx14rN5NbRJoWrn5LWIF4rGK1PCKnkxDdIs/  
zuFteYwSvy8eRjEpYimns2k+nSjvTV6kE9nJH2voerG/  
0+1uVkeGf0V3Dcn3eD3yc8ZAxjvXSzX8cVuzSMFYIdoLZyMHdzj0ACD0PuK8A0aa7srcIrtFggMvJAZcg4GQB

zngZ6fnoajrcxiUPChCrJk5+7kL2yODj88Z46VTowhOMle6atdtrotV8v60tnXk1WhR05J6S01tZPfp8uxn+M  
dUIjvbtLgw/ZoLmYN5fnAIoTd/  
FHg4PX24HGa+CPijPZatG91qMsLWMtv000rXarbWcE0oUM1wWmnlGMogj05vNy3+r2Yb6I8deLyLS9tLSV3d4  
ZEm2R+dvDzW8IjKjG3cJH0cn0zaAeo/Kr46/  
EgXPxF8Q+FidQuLzwl4d05NMu9KikivZ9QvIVkkumhTzDILHyhv3Db5wJlG8AcFV3xlnWVnU6Lyj/  
meHmlT2Mjpw50XW//AG71+eh5XYyRTaLqEUirmPvPvjngYXNjerG0nlbXQEYkF3n96PLP2fy1/1wf5f18/  
YBu00XV/  
B8t7cSW174zL8YaA2L25Ex0uG10sX9tdS3GV+1ecIZIkG8i1CbmcTPs2H8efA1kpmjBRvskUkEttbhtsM6hpi  
skIOTbtgjzUPnEkRgkFa/XL9lrTNKtPFxwkSaWXSbq/  
8AEWryNqrX0+KJZ9JmsLjaAxxhRH9tL7vOXftKbV+8r0eAxVWhWoqDjZTVrxv/  
AFsfa5LDC4jCV1JSu6btadv+fZ+meJv+eL5/35/+2UV1X/CvZf8AodrP/vtf/kiivrfr+I7x/wDAf6/  
p+h+ffUMP2n/4F/  
X9P0P5QPirrb3Ws+I0W5aXzvE2oX0UhXYfIbua0zRRubIKo7bgQB02nApvwM+IUfw1+NPw+  
+Js3h+Hxld+HNT1B5fCciXs3i6HUIIbc6LEz2d+9k7bA5vYr09MZ2g2jbgY8H4ovDe01yJPMG6e2DmMxNIsv1  
NIJmTc2zeLjaFLP8AcJ3H0Az4U+OpPhT8XPhj8XYdEh8V3Pw58Rf25H4WvLlbKy16P915tjc3bWmopAmKmp0  
2nXuzLDyDuzXz+SYXBfV6lSKm6lWDbUqjLfv3X8L7bn1MsXLCyXtG1NwftX3d0/8/8AM9j8aX8mqfEPV/  
ibZeGhofim38Ww+Jd08I6je3tqLa01mvdRh8HPLIFWNYbdrpl1IabHgRY0l/OPL/  
bP4X+LLLxP4A8E+PrK4sCnjzSk8QtHYHlitQ5FubEXDSv9sazuYbmEzrDaFvvNbpuAH4V+PviprvxX+I3j f4l  
a9p9lpN9468XXviG30Kz2vD4at7qCa3td0tpkhtVv1sre4u42uBaaeJ/  
PU+RDsw335+wB46t9V8Eal8JtaFzG3w1g1HU/  
BbTWxto59IvLxprpTpZTJJsfvMsUsNwTJ5huHRYUAzyUYwrvE05whyw0jyxUX9nqtep9Hluc1XGKU4811Zcqd  
9l/X4bH6sabrKyxAMHFAUZYlQM5yVyrBs4Hpt00ua3Y74R0sgWB0KSRvDLKI2feFctG/lv/  
AKvDZULYWHOAK4CyuFuFCRyGwQ+WqOE2MzPHG8i43chCyKrlsPy2F7bvn/ZPKuSsUwhDk+YpYKcLjgE5B2/  
XKjrXl1oRpV6cYXs2t9e3+Z9jgcVVqpSm4393aNT0r/mer+GbWzgs0MKyLerI0onjm01fNb03y/  
KIIG3rvXjPHIrtUN3Ess1wQ8UsEqzSm2eUMcfut5Rk2feYrndu04jG018myeP/  
AI1XM93F4J8JaTqFrbGNXmuNQisDiTf5BRZbKbfny5d2HG3C5+8K4vxH4n/  
ad1a0Wz1S3/4RuAscX6DqdpM14uRvW5c28Xl+SNvkEK2fNl90d+SpU9yk1GpJ2g2rpPzXU9NzrzXJScVUlPbt  
JpPzue7N8GtA1/  
VT4j1fSDd2tsb5p5Lk5aKmm4MRiN3HNv8A+eLmBV6fvQcZwsqw0C1tG+zadp9s1layXSxta28U2izbNouH81  
vML+X+7I2EeXJ3NfK974Y+MF88Nsde8W21vqRDT7vEKiKL70Vz+5htwJ/  
OE7ZDNH5YTgSFyUNR8GeKYcQ3XinXbFrRQIfsWot/  
pjpjzWuwMCQx7E8sHG3zZ0ea5q2WYxP6xWqQaprmajFLS0ezf8AXyv6GFwW0jT9vWq0pKn78lGCV9tFZ97/  
APDHuGu3Nrp0/  
wbkMkId2mbe8qkx8x5AjAcHbnJw3fuTXmt7d3M7ypDGGjQHexl2DDZC7fk0c7W6Y6cjmUtpCLKYbnVdavn6S  
Iksxfu3j0PM3uZ23ltsfJUY2nrnIv6ld29rGiGRmysm1BlQY0C+Y8knzBdoK7QV07ceQ0vN0oqUHUltdVnkY/  
EVYVoyglzRV17t+kenU4C5vbXTtcSW4e0hsoRPqmsPcMjBDpml7JrmdRswRF5qqylk3eYh3HZivx2j+LUmi/  
ET4neJtI0/Sdbh8W+JvGY06710xWWOLS9WXTTrfT7i1vM3G0Frcs0U8zzVwYvLy/  
6IfktFfi3+GPw58QDTYbRtS8faPq2gQ56hFvLitGW1S7nsZDn0RexGZV8vdiH940a/  
HDRfMS3sLct5wgt4Yw4Xy1PzSOSIwX2E7yD8zZ2jkly0KX1mtCuk3BSUm1stV8AgfPTU+WzbM4zw9eFWUXWlT  
ahb3dfdtP57pDbbo+jvAizI9mHkM0rCSWSQL5cZkd97G0HL+Wh3YCB2Aw0ew/  
Tn9mHWZL3xt4T0TU2t105U1K7DS0zJZ0mtYYUha1TKFSPtDGQq+SFTqTX5veALViYZSCdk0cY74PGc8Zx1I/  
Dmv0r/AGR9PL/F74X3hNqhuo/  
FNtLb3iCRYiINMkicAvGji3lsuwqmgz7m5WvfwCDxFFa61I9T4jnlP3ZNtNNP7L/8ij9H82P/Qbl/  
wDAGf8A+S6K7T+2bn/oG6R/34X/ABor6j2MP733/wDA/q/pbzfyw/vff/wP6v6W/  
kh1qK6eWRboATiGAYFV2Iz/  
ALwsRHn5M5+b5z05xxnjQ0kbbk+V4Dwxwc+Z144xt2jjJyDntXq+rwNIzncb3B0WxguuBtXGcDaBnqQc9BXmF  
78hl03GGHF73PX80nevjcvqTw06d0k3yXtaWun9JHvNWhp1HKtLm54K6tJpXtDddTdt7h5YwWGZHhLXc0GUGx  
gSKecbQxz1zn7wxX3F+wjPJH8f/EHgeS4mls/GXw0mktWuZizPq+juZLe1s/  
3WYBdi+mNw4dsC3i0yTAX8IaXdAKGMW791LFt3d5tgDfdxLPLzjGW3csMc/Xf7JWu3+h/tT/  
s8X1ibZV1DxTLo+rLcxiQ3GmzaVeCWNXJXy/mKsSvF7in0Ye3hoRVZpX/f00/w27bIMpvF0pJu91u32X+fu/  
bW0lm0u6/  
s3VXgju9sMzfZZhcQRWmgjXzAiB3BhLeHAGCMYBga6K5uVZVjWtZiz17buhHGTtxnvN0e0c1btPDs0pww1vF  
bRRBkvYYLwyB3eVNV1Jl1lchMpcCREij3YhEDESSTCef3U114e1qTR9djs3Tc0UrBnimTGcocKDNklvm+Xc

```
v3s14141MZU9r8NKWltLJcu9t/1PtcLmNej0Ebx5Lq94q9LZb/I9DsLwIF8yFDFb8RqTIN2/
r80bR42hBjIcYIC4wd2Rqs32lp3t7bcGUuyp9ofykjySxJnGeDnGAFpVwvMttqNpPKs09I9mdjBmx8+0hdq
k78HgfdAAOc89hp3g+w1NdPuJLuSL7f9pjFvbgNKEj8nzVuMyLTLiRRGSDyr4zjFdNSpQjCTpyaqJXi+br7vb
1f9NH1mFxcK0L83vaWa01uunfTtY+c77XrqKKCSyiu3ima4igliWVPNmi8pWiIZ5tu0yLtYHoTwTjGBBdX1xe
i08a5t74K73VnPG0ux0sJWUmPcJfnDDy12BV5b0R9ZX+ieHfDqDtD0synltJIGu2W42TnafNjG1QuSCXX0SQo
LdaDXC3g0vzEudQhtrzU55VtEmAW0WVJhhd+B0E+zgLGt5vmnlCuG8+piKzpyTm2rbN6dP8v60t6P1zEUQoUo
xn7ttnr209NEeBzLqE7PtB0WZY0PGNnU546kg4wMY5JPI8+8R3k8cM1p5ypJ0j2bnaCI7ibElvG8hdfJhkEFx
5kxUiPamVYnj1nxdrtpoS3kCRKJo5Z0EwL+SLGzGz92fMLL+RuXaFHXd8vxD8UdRvW8N+NtSu7m6ijm8K699L
eC4MDPMI4AJUfaxSWDcvLMASPNbGM15deTqQZ05/
DNWdtHa62fQ8fE1ZTTqS5SRStppuuh+bP7SvxVvfjN8SLj7Dqkp8CfD8SeF9IsGs/L3eIbcxR+KyJ/
OXz4N1rpBgupJKYhmwEKkt5t4csv0mC8hTsZcrnaE3cAFhn0B457c4ri9FtrhtJtHnneWaYyyyyFTmUSSsVkl
Xcd9w2D9ouN26f5MquwA+veFbUh4SepwJH97Pv6+v5jg162V040sPyRvaUb0+r2js3rc/
P8AMqcatdVJ0XNzJ2UmL9np8z60+H1rKbVDHD5mUDtzjaLPHQ5Eu/
r1XZzv3fL+jH70nh+Zvin80L5oJha2y3dxJPE5RLAXMMUbK67Sjv027hl4tohI58zK/
Fnw80Mw6fu+0bVaEhj5AZ1WTJLLH5w8zy9vzAFQdw+7X6u/speHhDYavr95uuLKw0qx0e4NttkkkjGp/
bpNu+Qw+VvtPk3N5hc/Mu3Ld0S1Z4nMVSqW5YVUo8qs9+rPKxn+z4atWp6TpwvG+qvdLVddz7C/4RbS/+g+f/
AABP/wAnUU3Fp6z/APfo/wCNf0X1a1L2l9/9f0/S3x/9q4vvd/wBf5/1f0t/LDqNi3lPuXB5x8v0zj5vb/
GvH9ZsSfN2nAz/AHSe/b5v519Bat/q/wAH/wDQVrxrWPuy/
wDAv51+U4etP21PVfEuh9tjYqrRq0fZbadYr9Dkd0jcFISpVbju/wBwjbb759c19M/
AG3YfHT4GajudTZeJr5s5jKxpmxEE+SbkJtLhLbQLtpXI5r5103/j4i+kv/stfU/wP/wCRx8E/9hr/
ANoXFe5Tr1YTjJSV019LeS/z+/yKymhTvSWtubv2UT9/
fA8i3eh6LHLG6LaLeW8+2feLlor+eW3vEmCL5Wf0cGLZKMAHz0cDoPFukW/
iSyW21LypLm13f2fd+WFhjhFz5kb4bM5by4drLkC7DlWDHHLFDH/kTPDP/Xvd/wDpQa7HWP8AWQ/V/
wD0BK8TF1Z060uW37347rvy7dj6b6tS/vf+BP8Ar/
hz5x1bSvEWh6tZahps9yx01blIrGCUw294JRBk3AAkVvK8oeWCj48x+Rk1rwfFua0W2g8Q2l3YXKL00Uum037
jb5fm+eohUSbz5ew7kC4k4bcSvbXv/H2f+B/0rxLxJ/yEr3/rLj/WuZVJLr/Wn+X9aW78E/
Y1acYX5eZaN3vp/
wAA607+PPHKOK8xDq2o3Xyo8szsGQpv2lFML43Z075hja0TXjetfFKLXniTT7q6hijulnuIJYmif92f3KRTbs
/NmXc3l/JheDnA8zf/AI+bv/ro/wD0qA/4+h9R/Om6s2mnaz8j2q9eao1GrXUdNPT/
AIP3+R3V7f3mquwuGaS1cyzRox30r0I9yvLgF921cHYoXBwDuIHzD+0zrD6N8HvEetY2paQS6R9h37CE1qa3g
E5uNjH90LbeIjbnzSceZGBk/Uft/qU+g/kk+Pf2yv8AKiXjP/r98P8A/pZJWdr2T2bX5nhYjGV/Y1NY/D/
Ku6Pzlt7JLS1ggVhJHbCOyWUJs80xQwzNKU3PsLfaQmzc+BHned2B6n4ThejRAL90BsdckZ+nPHU068585b/
AFQ/6/H/APSLTq9R8E/6wf8AXL/2Y17uJisFh6EqGjmlzcs5ui27Hzs37aSc+620tt/
kj7N+HRsayS48poJLRVebKpufM3n54jb/
ALrf5uwYbedhXlXzx+137PsVrD4S0iwsrdJH1G1huQkqiBYJwrmZHUh9hbevJILEYxxX4yfB7/X6Z/
19xf8AobV+0HwS+7H/ALi/+gy1nkEuXN8KraVq8VU13um902p5+bUoxy3GSV01SbXXW6PoX+yL7/nws/
8AwIT/AON0VYor9c9lT/lf/gT/AK7/ANLX8u5p/wA34Ly/4P3+R//Z"
```

## 7.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request

Headline	Headline description
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 7.5 Response example:

Click to expand...

```
{
  "FaceList": [
    {
      "quality": 0.756927,
      "eyeLeftX": 30.8689,
      "eyeLeftY": 48.5863,
      "eyeRightX": 57.8069,
      "eyeRightY": 48.1852,
      "faceRectLeft": 12,
      "faceRectTop": 32,
      "faceRectWidth": 64,
      "faceRectHeight": 64,
      "ProcessedPersons": 70,
      "ProcessedPersonsDuration": 0,
      "PersonList": [
        {
          "Id": "6C1069EB-1037-E711-9E4B-9C5C8E763A8F",
          "Name": "Hathaway",
          "Surname": "Anne",
          "Patronymic": "",
          "Department": "0",
          "Comment": "",
          "Timestamp": "2017-05-12T12:45:49.279",
          "ImageId": "6E1069EB-1037-E711-9E4B-9C5C8E763A8F",
          "AutoCreated": false,
          "LastRecognitionTimestamp": "2021-06-25T11:17:33.909",
          "RecognitionCount": 177,

```

```

"Sim": "0.54350495338439941"
}
]
}
]
}

```

## 7.6 Response parameters:

Parameter	Parameter description
FaceList	List of faces
quality	Face image quality (from 0 to 1)
eyeLeftX	Horizontal coordinates of the left eye
eyeLeftY	Vertical coordinates of the left eye
eyeRightX	Horizontal coordinates of the right eye
eyeRightY	Vertical coordinates of the right eye
faceRectLeft	Face coordinates relative to the left edge of the frame
faceRectTop	Face coordinates relative to the top edge of the frame
faceRectWidth	Face width in pixels

Parameter	Parameter description
faceRectHeight	Face height in pixels
ProcessedPersons	Number of generated vectors
ProcessedPersonsDuration	Time spent generating vectors
PersonList	A list of users with information from the faces database (see <a href="#">Get information about users from a specified department or the entire face database</a> )
id	User identifier from the faces database
Name	Name
Surname	Last name
Patronymic	Patronymic
Department	Department
Comment	Comment
Timestamp	Time of adding a face to the reference face database in UTC+0 format
ImageId	Identifier of the reference face image from the face database

Parameter	Parameter description
AutoCreated	<b>true</b> if the unrecognized face was added automatically
LastRecognitionTimestamp	Time of the last access in UTC+0 format
RecognitionCount	Number of accesses
Sim	Similarity level of the face on the image in the request between the reference face from the face database

## 8 Get an image by ID

### 8.1 General request format:

GET `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/firserver/GetImage/{server_id}/{id}`— get an image of a user from the face database by their ID in *Axxon PSIM*.

### 8.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>
id	Yes	ID of the reference face from the face database in <a href="#">GUID</a> format

### 8.3 Request example:

GET `http://1:1@127.0.0.1:10000/firserver/GetImage/1/7D02839C-CCB6-EA11-8E71-74D435D0E19B`

### 8.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request

Headline	Headline description
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 8.5 Response **example**:

JPEG image

## 9 Get analytical data

### 9.1 General request format:

POST `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/GetAnalytic`

### 9.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	Identifier of the Face recognition server in <i>Face PSIM</i>
minAge	No	Minimum age of persons If not specified, age is not taken into account
maxAge	No	Maximum age of persons If not specified, age is not taken into account
dateTimeFrom	Yes	Time in UTC+0 format starting from which you want to get the data
dateTimeTo	Yes	Time in UTC+0 format up to which you want to get the data
genders	No	Gender of persons: <b>0</b> –Unknown, <b>1</b> –Males, <b>2</b> –Females If not specified, gender is not taken into account
faceMode	No	Type of persons: <b>0</b> –all faces, <b>1</b> –only recognized faces, <b>2</b> –only unrecognized faces If not specified, then faceMode=0
threshold	No	Threshold value of the faces similarity level. The default value is 0.8 If not specified, then threshold=0.8

Parameter	Required	Parameter description
minPasses	No	Filter by the minimum number of accesses If not specified, filter is not taken into account
maxPasses	No	Filter by the maximum number of accesses If not specified, filter is not taken into account
cameraIds	No	Camera ID in <i>Face PSIM</i> If not specified, results for all cameras are returned

### 9.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/GetAnalyticyc

```
{
  "server_id": "1",
  "minAge": 0,
  "maxAge": 1000,
  "dateTimeFrom": "2018-08-17T21:00:00.000",
  "dateTimeTo": "2018-09-18T20:59:59.000",
  "genders": [
    0,
    1,
    2
  ],
  "faceMode": 0,
  "threshold": 0.8,
  "minPasses": 1,
  "maxPasses": 10000,
  "cameraIds": [
    "1"
  ]
}
```

## 9.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 9.5 Response **example**:

Click to expand...

```
{
  "PassAndLikesList": [
    {
      "MainPass": {
        "Id": "53952CAF-3BBB-E811-8D01-005056C00008",
        "Timestamp": "2018-09-18T12:09:32.848",
        "Camera": "1",
        "Gender": 1,
        "Age": 31.4502
      },
      "BestPerson": {
        "Id": "5CC71799-48BB-E811-8D01-005056C00008",
        "Name": "John",
        "Surname": "Wick",
        "Patronymic": ""
      }
    }
  ]
}
```

```

"Department": "Department 1",
"Comment": "",
"Timestamp": "2021-06-25T10:39:55.974",
"ImageId": "98B73FA9-A1D5-EB11-AC0A-B42E999B6711",
"AutoCreated": false,
"LastRecognitionTimestamp": "2021-06-25T11:59:52.185",
"RecognitionCount": 375,
"revisions": [
  {
    "source": "8A91CD97-62D3-EB11-AC0A-B42E999B6711",
    "revision": "97B73FA9-A1D5-EB11-AC0A-B42E999B6711"
  }
]
"Sim": 0.850095
},
"LikePasses": [
  {
    "Id": "1E1483A5-41BB-E811-8D01-005056C00008",
    "Timestamp": "2018-09-18T12:52:08.987",
    "Gender": 1,
    "Age": 31.4372,
    "Camera": "1"
  },
  {
    "Id": "F10CD78D-41BB-E811-8D01-005056C00008",
    "Timestamp": "2018-09-18T12:51:33.907",
    "Gender": 1,
    "Age": 31.4372,
    "Temperature": 0,
    "Emotion": 5,
    "Glasses": 2,
    "FacialHair": 6,
    "HairColor": 1,
    "HairType": 3,
    "Headwear": 13,
    "Liveness": 1,
    "Evasion": 1
  }
],
"Total": 40,
"Males": 40,
"Females": 0
},
"ProcessedTotal": 1346,
"ProcessedDuration": 83,
"CompareCount": 18555,
"Response": {
  "Status": "SUCCESS"
}
}

```

## 9.6 Response parameters:

Parameter	Parameter description
PassAndLikesList	List of accesses of unique faces
MainPass	Data on the first access of a unique face (for more details, see <a href="#">Get the recognition protocol</a> )
BestPerson	Data about the face from the face database and the similarity level of the reference face between a unique face, if such a person exists (for more details, see <a href="#">Get information about users from a specified department or the entire face database</a> )
Timestamp	Time of adding a face to the reference face database in UTC+0 format
ImageId	Identifier of the reference face image from the face database
AutoCreated	<b>true</b> if the unrecognized face was added automatically. <b>False</b> if it was added manually
LastRecognitionTimestamp	Time of the last access in UTC+0 format
RecognitionCount	Number of accesses
LikePasses	List with data of similar accesses (similar to MainPass). Cannot be greater than 5 elements
Temperature	Face temperature in degrees Celsius

Parameter	Parameter description
Emotion	Emotion: 0–undefined, 1–rage, 2–fear, 3–happiness, 4–without emotion, 5–sadness, 6–astonishment
Glasses	Glasses: 0–undefined, 1–tinted glasses, 2–without glasses, 3–glasses
Facial Hair	Facial hair: 0–undefined; 1–beard, 2–bristle, 3–imperial beard and mustache, 4–imperial beard, 5–mustache, 6–absent
HairColor	Hair color: 0–undefined, 1–brunet, 2–blonde, 3–chestnut hair, 4–ginger, 5–gray
HairType	Type of bald head: 0–undefined, 1–bald, 2–with receding hairline, 3–without bald head
Headwear	Headwear: 0–undefined, 1–cap, 2–bandana, 3–beret, 4–beanie, 5–trapper hat, 6–fur hood, 7–glasses on head, 8–hat, 9–helmet, 10–hood, 11–kepi, 12–headscarf, 13–without headwear, 14–peaky cap
Liveness	Artificial face: 0–undefined, 1–no, 2–yes
Evasion	Face concealment: 0–undefined, 1–face not covered, 2–mask is on, 3–tinted glasses, 4–face partially covered, 5–balaclava, 6–other mask, 7 – mask isn't on properly
Total	Total number of unique faces
Males	Number of males
Females	Number of females
ProcessedTotal	Total number of processed faces
ProcessedDuration	Execution time of the faces processing in milliseconds

Parameter	Parameter description
CompareCount	Total number of the face comparisons
Response	Response
Status	Status of the completed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>

## 10 Get analytical data by time

### 10.1 General request format:

POST `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/GetAnalytcByTime`

### 10.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	Identifier of the Face recognition server in <i>Face PSIM</i>
minAge	No	Minimum age of persons If not specified, then age is not taken into account
maxAge	No	Maximum age of persons If not specified, then age is not taken into account
dateTimeFrom	Yes	Time in UTC+0 format starting from which you want to get the data
dateTimeTo	Yes	Time in UTC+0 format up to which you want to get the data
cameras	No	Camera IDs in <i>Face PSIM</i> If not specified, then results for all cameras are displayed
genders	Yes	Gender of persons: <b>0</b> –Unknown, <b>1</b> –Males, <b>2</b> –Females If parameter value is empty, then all results are displayed regardless of gender

Parameter	Required	Parameter description
returnLocalTime	No	Time format of the face capture: <b>0</b> —time in UTC+0 format, <b>1</b> —local time of the server If not specified, then returnLocalTime=0
timePeriod	No	Type of grouping of analytical data: <b>0</b> —by hour, <b>1</b> —by day, <b>2</b> —by month, <b>3</b> —by year If not specified, then timePeriod=0

### 10.3 Request example:

POST http://1:1@127.0.0.1:10000/firserver/GetAnalytcByTime

```
{
  "server_id": "1",
  "minAge": 20,
  "maxAge": 30,
  "dateTimeFrom": "2019-02-25T21:00:00.000",
  "dateTimeTo": "2019-03-29T20:59:59.000",
  "cameraIds": [
    "1"
  ],
  "genders": [
    0,
    1,
    2
  ],
  "returnLocalTime" : 0,
  "timePeriod" : 0
}
```

### 10.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request

Headline	Headline description
Content-Type	Returns the type of received request
Server	Specifies the server from which the request comes
Access-Control-Request-Method	Returns the HTTP method that can be used in the request
Access-Control-Origin	Shows if the server response can be accessed by the code that sends a request from this source
Date	Shows the time of the request

## 10.5 Response **example:**

Click to expand...

```
{
  "Rows": [
    {
      "timestamp": "2019-02-27T180000.000",
      "value": [
        {
          "age": 27,
          "gender": {
            "males": 10,
            "females": 9,
            "unknown": 0
          },
          "smile": {
            "males": 3,
            "females": 1,
            "unknown": 0
          },
          "angry": {
            "males": 1,
            "females": 2,
            "unknown": 0
          },
          "fear": {
            "males": 1,
            "females": 2,
            "unknown": 0
          }
        }
      ]
    }
  ]
}
```

```

    "happiness": {
      "males": 1,
      "females": 1,
      "unknown": 0
    },
    "neutral": {
      "males": 2,
      "females": 1,
      "unknown": 0
    },
    "sadness": {
      "males": 1,
      "females": 1,
      "unknown": 0
    },
    "surprise": {
      "males": 1,
      "females": 1,
      "unknown": 0
    }
  }
]
}
"Response": {
  "Status": "SUCCESS"
}
}

```

## 10.6 Response parameters:

Parameter	Parameter description
Rows	List of unique faces
timest amp	Time of first access in UTC+0 format
value	List of additional face characteristics (for more details, see <a href="#">Get the recognition protocol</a> )
Respo nse	Response

Parameter	Parameter description
Status	Status of the completed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>

## 11 List of possible errors of the Face recognition server

### 11.1 General request format:

POST http://AxxonPSIM\_user\_login:AxxonPSIM\_user\_password@IP address:port/firserver/  
FaceErrorDescriptions

### 11.2 Request example:

POST http://1:1@127.0.0.1:10000/firserver/FaceErrorDescriptions

### 11.3 Response:

```
{
  "FAILED_TO_ADD_PERSON": "Failed to add person. Check request parameters or database",
  "FACE_NOT_FOUND": "Face not found on supplied image",
  "MAXIMUM_FIRS": "Maximum FIR in database",
  "SERVER_NOT_READY": "Server is initializing, please wait",
  "FAILED_TO_DELETE_PERSON": "Failed to delete person. Check request parameters or
database",
  "FAILED_TO_ADD_PERSON_MANY_FACES": "The image shown has more than one face",
  "FAILED_TO_SAVE_DATABASE": "Failed to save person. Check request parameters or
database",
  "FAILED_TO_UPDATE_PERSON": "Failed to update person. Check request parameters or
database"
}
```

### 11.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request

Headline	Headline description
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 11.5 Response parameters:

Parameter	Parameter description
FAILED_TO_ADD_PERSON	Failed to add a person. Check the request parameters or database
FACE_NOT_FOUND	Face not found on the supplied image
MAXIMUM_FIRS	Maximum number of faces in the database
SERVER_NOT_READY	Server is initializing, please wait
FAILED_TO_DELETE_PERSON	Failed to delete a person. Check the request parameters or database
FAILED_TO_ADD_PERSON_MANY_FACES	The image has more than one face
FAILED_TO_SAVE_DATABASE	Failed to save a person. Check the request parameters or database
FAILED_TO_UPDATE_PERSON	Failed to update a person. Check the request parameters or database

## 12 Face search by camera and time

### 12.1 General request format:

POST `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/FindPasserBy`

### 12.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	Identifier of the Face recognition server in <i>Face PSIM</i>
data	Yes	Data you want to get
cameralds	No	Camera identifiers (GUID). If they are not specified, the results for all cameras are displayed
dateTimeFrom	Yes	Time in UTC+0 format starting from which the face search is performed
dateTimeTo	Yes	Time in UTC+0 format up to which the face search is performed
sim	No	Similarity level between the face from the camera photo and the reference image in the request in the range of 0–1. The recommended value is 0.8
image	Yes	Image in base64 format

### 12.3 Request example:

POST `http://1:1@127.0.0.1:10000/firserver/FindPasserBy`

```
{
```

```

"server_id" : "1",
"data":
{
"cameraIds" : ["1"],
"dateTimeFrom" : "2023-08-23T12:20:00",
"dateTimeTo" : "2023-08-25T16:20:00",
"sim" : 0.9,
"image" : "Alex's photo in base64"
}
}

```

## 12.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request. Measured in bytes
<b>Content-Type</b>	Returns the type of the received request
<b>Server</b>	Specifies the server from which the request comes
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 12.5 Response **example**:

**Click here to expand**

```

{
  "Data": {
    "rows": [

```

```

{
  {
    "id": "301F1817-2B43-EE11-B47A-D85ED323673C",
    "sim": 0.994588,
    "cameraID": "aaffb7ea-2743-ee11-b47a-d85ed323673c",
    "timestamp": "2023-08-25T09:38:08.027"
  },
  {
    "id": "7F1D0A07-2B43-EE11-B47A-D85ED323673C",
    "sim": 0.993709,
    "cameraID": "aaffb7ea-2743-ee11-b47a-d85ed323673c",
    "timestamp": "2023-08-25T09:37:47.249"
  },
  {
    "id": "CA87811E-2B43-EE11-B47A-D85ED323673C",
    "sim": 0.993194,
    "cameraID": "aaffb7ea-2743-ee11-b47a-d85ed323673c",
    "timestamp": "2023-08-25T09:38:18.267"
  }
],
  "totalCount": 3,
  "duration": 127
},
  "Status": "OK"
}

```

## 12.6 Response parameters:

Parameter	Parameter description
Data	Received data
rows	List of accesses of the face captured by the cameras
id	Identifier of the user from the faces database
sim	Similarity level between the face in the camera photo and the reference image in the request. The results must be equal to or greater than the specified value
camerald	GUID of the camera that captured the face
timestamp	Time of face capture/access in UTC+0 format

Parameter	Parameter description
totalCount	Number of accesses of the face
duration	Face processing time in milliseconds
Status	Status of the executed request: SUCCESS—successful, if error—see <a href="#">List of possible errors of the Face recognition server</a>

## 13 Number of faces in the frame

### 13.1 General request format:

GET `http://AxxonPSIM_user_login:AxxonPSIM_user_password@IP address:port/firserver/GetNumberFacesInFrame?server_id=server ID&capture_id=face detector ID`—get information about the number of faces in the frame.

### 13.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	Identifier of the Face recognition server in <i>Face PSIM</i>
capture_id	Yes	Identifier of the face detector in <i>Face PSIM</i>

### 13.3 Request example:

GET `http://1:1@localhost:10000/firserver/GetNumberFacesInFrame?server_id=1&capture_id=1.1`

### 13.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request. Measured in bytes
<b>Content-Type</b>	Returns the type of the received request
<b>Server</b>	Specifies the server from which the request comes

Headline	Headline description
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

### 13.5 Response example:

```
{
  "NumberFacesInFrame": 3
}
```

### 13.6 Response parameters:

Parameter	Parameter description
NumberFacesInFrame	The number of faces in the frame of a specific camera at the current moment in time

## 14 Get a list of all departments

### 14.1 General request format:

POST http://AxxonPSIM\_user\_login:AxxonPSIM\_user\_password@IP address:port/firserver/ReadDepartments

### 14.2 Request parameters:

Parameter	Required	Parameter description
server_id	Yes	ID of the Face recognition server in <i>Face PSIM</i>

### 14.3 Request example:

POST http://1:1@127.0.0.1/firserver/ReadDepartments

```
{
  "server_id": "1",
}
```

### 14.4 Request headlines:

Headline	Headline description
<b>Allow</b>	Returns the HTTP method used in request
<b>Content-Length</b>	Shows the size of characters in bytes received in the request
<b>Content-Type</b>	Returns the type of received request
<b>Server</b>	Specifies the server from which the request comes

Headline	Headline description
<b>Access-Control-Request-Method</b>	Returns the HTTP method that can be used in the request
<b>Access-Control-Origin</b>	Shows if the server response can be accessed by the code that sends a request from this source
<b>Date</b>	Shows the time of the request

## 14.5 Response example:

[Click here to expand](#)

```
{
  "Departments": [
    {
      "Id": "1",
      "Name": "1",
    }
  ],
  "AllowCreateDepartments": true,
  "Status": "OK"
}
```

The response returns a list of all departments.

## 14.6 Response parameters:

Parameter	Parameter description
Departments	List of departments
Id	Department ID
Name	Department name
AllowCreateDepartments	Whether it is allowed to create departments

Parameter	Parameter description
Status	Status of the completed request: OK—successfully, in case of error—see <a href="#">List of possible errors of the Face recognition server</a>