**1 Overview**

**The Role of This Platform:**

This platform is available to users who need to do advanced development on the basis of SimCam cameras. On the SimCam open platform, users can configure the address of their own server, and the related events notifications of the SimCam camera will be given back to the user server for advanced development. At the same time, users can develop applications based on the API documentation to modify the configuration of the SimCam camera.

**Targeted Crowd:** Users registered in the SimCam App.

This document mainly introduces functions of the SimCam open platform and the API quick start guide. If you have any further questions, please contact us at support@simcam.ai or sdk@simcam.ai.
2 Introduction of SimCam Open Platform

2.1 Introduction of Login Interface

Address:  https://platform.simcam.cn/

Prompt: Please login the platform by the account and password registered in SimCam APP. If you don't have a SimCam App, please download the App and proceed with the registration.

2.2 Introduction of Camera Configuration Interface
2.2.1 Camera List Configuration

**Description**: The lists of devices that users have paired with the Simcam App.

**Q**: What if there is no device in the interface of camera configuration?

**A**: Pair a Simcam camera to the SimCam App.

2.2.2 Device name

**Description**: Device's name that has been paired to the Simcam App.

**Q**: How to change the camera's name?

**A**: Change camera's name on Simcam app and then refresh the open platform.

2.2.3 Service address

**Description**: The server's address that provides notifications for users.

**Requirements of The Format**: (domain name + port + method path) or (ip + port + method path)

For Example:
- http://www.baidu.com/simcam/receiveEvent

Feedback Types of Events Notifications Supported by The Open Platform

- Person Detected
- Object Monitoring
- Face Recognized

Definition of Event Notifications Interface Received by The Server

- **Receiving Method**: POST
- **Parameter Type**: form-data (x-www-form-urlencoded)
- **Interface Parameter**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>devicename</td>
<td>Device Name</td>
<td>String</td>
<td>/</td>
<td>YES</td>
</tr>
<tr>
<td>eventType</td>
<td>EventType</td>
<td>Int</td>
<td>/</td>
<td>YES</td>
</tr>
<tr>
<td>eventTime</td>
<td>EventTime</td>
<td>Time Stamp</td>
<td>10</td>
<td>YES</td>
</tr>
<tr>
<td>payload</td>
<td>payload</td>
<td>String</td>
<td>/</td>
<td>NO</td>
</tr>
</tbody>
</table>

- **Introduction of eventType**
<table>
<thead>
<tr>
<th>Value</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Int</td>
<td>Person Detected</td>
</tr>
<tr>
<td>2</td>
<td>Int</td>
<td>Object Monitoring</td>
</tr>
<tr>
<td>3</td>
<td>Int</td>
<td>Face Recognized</td>
</tr>
</tbody>
</table>

- **Introduction of payload**
  - When the `eventType` type is 1 or 2, there is no `payload` parameter.
  - When the `eventType` type is 3, `payload` is the current face ID recognized by the camera.

### 2.2.4 Enabled
**Description**: Switch on event notifications. Once turned on, if the camera detects an event, it will send an event notification to the server set by the user, and if the event notifications is off, event notification will not be sent.

### 2.2.5 Set
**Description**: Configuration interface of a single camera's event. Enter the server address where the user receives event notifications in **Server address**.

**Specs**

**Device name**: 1-SimCam

**Server address**: [Enter server address]

**Enabled**: [Switch]

After configuration, click the **test** button, and we will send an HTTP request to your server as follows:

For example: `http(https)://www.google.com/...`
If the communication is normal, we will be prompted:

**Tips:**

```
success: 200 OK
```

**CONFIRM**

### 2.2.6 CONFIGURE

**Description:** Configure event notification addresses for all cameras together.

### 2.3 Introduction of Developer Interface
**Description:** Click key to generate client_id and client_secret. client_id and client_secret can be used to get access_token. access_token can be used to call related API interface. Each account has a unique client_id and client_secret, please do not disclose them.

**Prompt:** Client_secret can only be viewed after clicking the key. It will not be displayed after refreshing the interface or re-login. Please keep it safely.

**Q:** Why is my Secret key like ****************************

**A:** The Secret key is encrypted and cannot be decrypted. If you have forgotten the Secret key, please click KEY again to regenerate a Secret key for you, please keep it safely.

### 3 SimCam platform: API Introduction

This section introduces the Simcam open platform API interface. Users can manage this API to configure the camera. At present, only the API for configuring the camera privacy mode is available, and other APIs will be available in accordance with requirements.

#### 3.1 Interface Description

##### 3.1.1 Request Data

**Post Request Message**

```plaintext
POST /XXX/XXXXXXXX HTTP/1.1
Host: platform.simcam.cn
Content-Type: application/x-www-form-urlencoded
Authorization: Bearer 077e604e-596c-44db-9a5d-41c3c867494f
```

Note: All interface request headers need to be placed in the Authorization property, the value is token_type + access_token, except for taking token's interface. There is a space between token_type and access_token.

##### 3.1.2 Request Return

Return data: If the returned HTTP status code is 200, then back to the json data, otherwise, it means a request error. Please estimate it according to the interface return code.

### 3.2 Certification Management

#### 3.2.1 Get Token

1. Function Description

   The third-party platform authenticates to the Simcam open platform and requests an access token. The access_token is valid for 43199 seconds. During this valid period, it doesn't need to be required repeatedly during the validity period and can be reused.
2. How to Obtain `client_id` and `client_secret`: Login to the open platform through the Simcam App account ([https://platform.simcam.cn/](https://platform.simcam.cn/)). For detailed operation steps, please refer to document 2.3.

3. Interface Definition

1. Interface address: [https://platform.simcam.cn/oauth/token?grant_type=client_credentials](https://platform.simcam.cn/oauth/token?grant_type=client_credentials)

2. Request method: post

3. Interface parameters

   1. Interface address:

      [https://platform.simcam.cn/oauth/token?grant_type=client_credentials](https://platform.simcam.cn/oauth/token?grant_type=client_credentials)

   2. Request method: **POST**

   3. Interface parameters

      Note: `client_id` and `client_secret` are transmitted in Basic Auth encryption with postman as an example:

      | Attribute Name | Attribute Description | Type | Length | Required |
      |----------------|-----------------------|------|--------|----------|
      | client_id      | Client ID (Match to the last picture ‘Username’) | String | 128    | YES      |
      | client_secret  | Client Secret (Match to the last picture ‘Password’) | String | 256    | YES      |

2. HTTP request message

```
POST /oauth/token?grant_type=client_credentials HTTP/1.1
Host: platform.simcam.cn
Authorization: Basic
d2h4Gy2YTlOGywMwI4YTRjN2I6YTU4OWRiZTM3MzBhNGEzY2FmYjkyMTU4ODFhYmM0NW4=
User-Agent: PostmanRuntime/7.15.2
```
### 3.3 Equipment Management

#### 3.3.1 Get Device List

1. **Function Description**
   
   Get all equipment information under user's account.

2. **Definition of the Interface**

   1. Request address:  `https://platform.simcam.cn/api/getDeviceList`
   2. Request method:  `POST`
   3. Request message

   ```
   POST /api/getDeviceList HTTP/1.1
   Host: platform.simcam.cn:8081
   Content-Type: application/x-www-form-urlencoded
   Authorization: Bearer ca250c38-350f-478b-8807-6c2fb8ce0257
   ```

4. **Output parameters** (json)

   Operate successfully and return:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>cameras</td>
<td>see the list below</td>
<td>list</td>
<td>/</td>
<td>YES</td>
</tr>
</tbody>
</table>

1. **Cameras attribute introduction:**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>device_id</td>
<td>Device's id</td>
<td>String</td>
<td>/</td>
<td>YES</td>
</tr>
<tr>
<td>device_name</td>
<td>User-defined device name</td>
<td>String</td>
<td>/</td>
<td>YES</td>
</tr>
</tbody>
</table>
3.4 Camera Configuration Management

3.4.1 Setting the Privacy Mode of The Camera

1. Function Description
   Setting the privacy mode of camera

2. Definition of Interface

   1. Request address: [https://platform.simcam.cn/api/deviceControl](https://platform.simcam.cn/api/deviceControl)
   2. Request method: POST
   3. Request parameter

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>device_id</td>
<td>Device's Id</td>
<td>String</td>
<td>30</td>
<td>YES</td>
</tr>
<tr>
<td>mode</td>
<td>Mode selection, see table below</td>
<td>String</td>
<td>20</td>
<td>YES</td>
</tr>
<tr>
<td>mode_switch</td>
<td>Switch (1:On 0:Off)</td>
<td>int</td>
<td>2</td>
<td>YES</td>
</tr>
</tbody>
</table>

1. Mode Parameter Introduction
   At present, we only support privacy mode switch, and we will update it into the file if a new function is added.

<table>
<thead>
<tr>
<th>Parameter Values</th>
<th>Parameter Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>YSMS</td>
<td>Privacy Mode</td>
</tr>
</tbody>
</table>

3. Http Request Message

   ```
   POST /api/deviceControl HTTP/1.1
   Host: platform.simcam.cn:8081
   Content-Type: application/x-www-form-urlencoded
   Authorization: Bearer 94816501-fbc7-4a36-9b41-622bc0a786ed
   ```

4. Output parameters (json)
   Operate successfully and return:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>success</td>
<td>String</td>
<td>10</td>
<td>YES</td>
</tr>
</tbody>
</table>
操作失败返回参数（json）

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Description</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>error</td>
<td>String</td>
<td>10</td>
<td>YES</td>
</tr>
<tr>
<td>data</td>
<td>Detailed Error Information</td>
<td>String</td>
<td>/</td>
<td>YES</td>
</tr>
</tbody>
</table>