

N3N Camera API

Summary:

N3N Camera API is for video streaming and using PTZ controls.

Description:

We use our lightweight N3N Camera API's for below function areas:

1. Camera List API

- Metadata
 - To get the Camera Name, ID, RTSP url, Status and Geo Coordinates.
- Request and Response

GET:

```
/media/cameraInfo?token=[PublicAccessToken]&name=[CameraName]
```

Response: [

```
  [Camera Name]: {
    "name": "[Camera Name]",
    "description": "[Camera Description]",
    "label": "[Camera Display Name]",
    "mediaserver": "[Media Server Name]",
    "status": "[Status]",
    "mediaLink": "[Camera Link]",
    "properties": {
      "rtsp": "[Camera RTSP URL]",
      "RtspID": "[RTSP Login ID]",
      "RtspPassword": "[RTSP Login Password]",
      "latitude": "[Geo Location]",
      "longitude": "[Geo Location]",
      "PlayerTypeName": "video",
      "UsePTZ": "[PTZ Support]"
    },
    "metrics": [Information from VSM]
  },
]
```

2. Set Camera API

- Add new Camera to the List
 - To dynamically add new camera to the camera based on Geo-Coordinates
- Request and Response

POST: /media/setcamera

Body:

```
{
  "token": [API Access Token],
  "name": [Unique Name],
  "description": [Description],
  "label": [Display Name],
  "rtsp": [RTSP URI],
  "RtspID": [RTSP Login ID],
  "RtspPassword": [RTSP Login Password],
  "latitude": [Geo Location],
```

```
"longitude": [Geo Location],  
"UsePTZ": [If this Camera support PTZ, true or false]  
}
```

3. Image Capture (see addendum for code)

- Save as Image (JPEG) locally
 - To Capture the video stream as Image and Save
 - `mp1.saveAsImage("c:/test.jpeg");`

4. Video Capture (see addendum for code)

- Save as Video file (AVI) locally
 - To Record and save the video file
API:
 1. Start the video capture
 - `mp1.startSaveAsVideo("c:/test.avi");`
 2. End the video capture
 - `mp1.endSaveAsVideo("c:/test.avi");`

5. Video Render SDK

- Single Camera rendering using Javascript
 - To dynamically render a single video.
 - ```
document.addEventListener('DOMContentLoaded',
function() {
 mp1 = new N3N.WizeyeVideoPlayer();
 mp1.open("video",
 "http://cdp.w20.n3n.io", "cam1",
 "1504878079591:607602");
});
```
- Full Screen rendering
  - To dynamically use the url as new window popup with supporting PTZ controls.  
<http://<serverip>/view?token=1506497045480:997261&mapId=a0ac4dc2-6924-bcf8-4213-b9facdba3a59&objectName=CCTV-P03-001>

#### Pre-requisite:

- Need to install MD server and Wizeye software.

#### Technical Requirements:

1. Install MD for video streaming and Wizeye for PTZ controlling
2. MD Server must be installed windows OS
3. Access to IP Cameras from MD Server

**Data Collection Interval:**

- Streaming API:
  - Through RTSP when the client request the Video.

**Latest N3N API Version tested:**

- N3N 2.0

## Addendum

```
<html>
<body style="position: absolute;top:0; bottom:0;right:0;left:0;overflow:hidden;">
 <div id="video" style="width:1280px; height:720px"></div>
 <div id="message" style="width:1280x; height:100px"></div>
 <input type="text" id="StillImageFilePath" size="20"></input><button
onclick="saveAsImage()">STILL</button><p>
 <input type="text" id="VideoFilePath" size="20"></input><button
onclick="startSaveAsVideo()">VIDEO</button><p>
 <button onclick="endSaveAsVideo()">END</button><p>
 <script type="text/javascript"
src="http://cdp2.w20.n3n.io/public/mediaplayer/mediaplayer.js"></script>
 <script type="text/javascript">
 var mpl;
 document.addEventListener('DOMContentLoaded', function() {
 mpl = new N3N.WizeyeVideoPlayer();
 mpl.open("video", "http://cdp2.w20.n3n.io", "cam1", "1506546503490:667375");
 mpl.onEvent = function(json)
 {
 var message = document.querySelector("#message");
 message.innerHTML = JSON.stringify(json);
 }

 document.querySelector("#StillImageFilePath").value = "e:/test.jpg";
 document.querySelector("#VideoFilePath").value = "e:/test.avi";
 });
 window.onunload = function() {
 if (mpl) mpl.close();
 };

 function saveAsImage()
 {
 mpl.saveAsImage(document.querySelector("#StillImageFilePath").value);
 }

 function startSaveAsVideo()
 {
 mpl.startSaveAsVideo(document.querySelector("#VideoFilePath").value);
 }

 function endSaveAsVideo()
 {
 mpl.endSaveAsVideo();
 }
 </script>
</body>
</html>
```