USER MANUAL

This user manual is designed to ....

Date: 8th August 2016

Version: 1.3

# COPYRIGHT AND LEGAL NOTICE FOR DOCUMENTATION

This documentation is provided for informational purposes only, and nViso and its suppliers make no warranties, either express or implied, in this documentation.

Information in this documentation, including URL and other Internet Web site references, is subject to change without notice. The entire risk of the use or the results of the use of this documentation remains with the user.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user.

Without limiting the rights under copyright, no part of this documentation may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of nViso.

nViso may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this documentation. Except as expressly provided in any written license agreement from nViso, the furnishing of this documentation does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2016 nViso SA. All rights reserved.

# JAVA SDK SAMPLE

# Getting Started

You first need to download the source code for Java SDK at https://github.com/nViso/developer-samples. The package should contain two folders corresponding to the Console and the GUI Applications as well as two folders named “maven” and “jre1.8.0\_73” that you will need in order to build these projects.

# Requirements

To run the application you will need the Java JRE 1.8 or higher installed. We provide all the files of JSE-1.8 in the folder jre1.8.0\_73.

http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html

# Console Application

## Requirements

Building the API client library requires [Maven](<https://maven.apache.org/>) version 2.2 to be installed. We provide all the required files in the folder maven, but you will need to add the path to its bin folder to the environment variable “Path”. Check that everything went fine by typing in a terminal : mvn --version.

## Compilation

In order to install the API client library to your local Maven repository, simply execute: mvn install. Add option -DskpipTests if you do not want to execute the tests at the same time.

## How to Use

Once the dependencies have been installed, you can execute the tests from the terminal with : mvn test

This will run all the tests, which are located in src\test\java\io\swagger\client\api. There are five tests that are run :

* healthTest in MonitoringApiTest.java which just tests the health of all workers. It should print "info: All the workers responded to the PING."
* photoProcessUploadTest in PhotoApiTest.java which processes a .jpg photo using photoProcessUpload method. The program then extract all emotion intensities from the json and prints the emotion with the highest intensity.
* streamProcessUploadTest in StreamApiTest.java which starts a session and processes a stream of .jpg photos using streamProcessUpload method. The program then ends the session, performs a query to get the result in json format and extract all emotion intensities for all images and prints each time the emotion with the highest intensity.
* streamSessionDataTest in StreamApiTest.java which does the same as streamProcessUploadApiTest but just prints the resulting json object in csv format using streamSessionData method.
* videoProcessUploadTest in VideoApiTest.java which upload and processes a video, and prints for each frame the emotions of each face present on the frame.

# GUI Application

## Requirements

If you want to recompile the project, you will need Eclipse to be installed. Otherwise, we provide a runnable .jar file from which you can execute the project.

## Compilation

In order to compile the cource code, you can use Eclipse. In that case, download the source code and import the project in Eclipse. Then you need to add the .jar libraries to the build path. To do so, right click on the project and select : Build Path -> Configure Build Path... Then click on “Add JARs” and add all the files inside the lib folder. Then you can run the project as a Java Application.

If you just want to execute the project, you can do it from the runnable .jar file located in the bin folder. Open a terminal in this folder an execute command : java -jar NVisoGUI.jar SESSION\_ID SESSION\_KEY, where SESSION\_ID and SESSION\_KEY are optional parameters indicating the id and the key of the session.

## How to Use

The interface is very easy to use, it contains 3 buttons :

* SELECT FOLDER which enables you to upload the folder containing the files you want to process (only selects files with .jpg, .jpeg or .png extension). Once you have selected a set a files, they are displayed in a tab and are marked as "Not processed". You can click on a file to display the corresponding image.
* PROCESS which enables you to process all the files currently loaded. This is done asynchronously and you can observe the work progression in the progress bar. Once the processing for a file is finished, it is marked as "Processing OK" or "Processing failed" depending on wether the processing has succeeded or failed. The images for which the processing succeeded are displayed with red dots corresponding to the landmarks used during the processing. Just under the image is displayed the json returned by the process.
* EXTRACT which enables you to upload the processing results in a file in csv format. By clicking on select you can specify a folder to process. The application will search all image files in that folder with extension .jpg, .jpeg, or .png. It will then process the images and you can export the results as a CSV file.

## CSV Format

The generated csv file contains several informations about the images that have been processed, namely :

* The date at which the file has been processed.
* Intensities of all the different emotions of the processed face, ie Neutral, Sadness, Disgust, Anger, Surprise, Fear and Happiness.
* Indicates whether or not a face has been found.
* Some message if an error has occured during the processing.

# Trouble Shooting



**nViso SA**

Building D, Innovation Park, EPFL

CH-1015 Lausanne,

Switzerland

T: +41 21 693 85 11

info@nviso-insights.com

www.nviso-insights.com