das-Face v3.0

Release 2021Q1

Release	Date	Description	Written	Reviewed	Aproved
1	11/02/2021	First version	DGS	JGC	JGC

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

1 What's new?

This version comes with two major changes related to biometric models and antispoofing techniques.

- New biometric model which reduces false positive and false negative rates, and improves the accuracy on different demographic cases (Caucasian, African, Indian, Asian). The model improves the accuracy of selfie-mode from 99.8% to 99.9% and the funnel of document-mode is improved by two points maintaining the same security level, obtaining TPR=2.43% @ FPR=0.62% at 0.7 threshold. The new model is identified by:
  - Model hash: 904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15
  - Model tag: 20210203
- Because model 20210203 generates larger biometric credentials than earlier models, the default endpoint /credential/photo will generate them using model 20200514, so by default credentials have the 2896 bits length as usual.
- When security is a must, and having a use case that could afford using larger biometric credentials, the model 20210203 is available at the endpoint:
  - /models/<hash>/<mode>/credential/photo.

  The biometric credentials for model 20210203 are of 9040 bits length.
- An update of the Selfie-Alive Pro (SAP) use case (/challenges/analysis/video-photo) decreasing the time of the operation in a 26%. The system is calibrated at threshold 0.7 for BPCER=2.7% and APCER=0.9%.
- An update of the passive AS solution (/authenticity/photo) calibrated at threshold 0.7 for BPCER=3.2% and APCER=7.8%, reducing a 50% the APCER of the previous das-Face version. The system requires additional 300ms per query (a total of 1185ms per query).
- **Deprecation announcement**: The biometric model with:
  - model hash: 3a9e9d5ffd5de4c212c2aff26eeca523fb69754e604894520b32e4ed
  - model tag: 20190813

  is marked as deprecated, enforcing deprecation for the **2021 Q3** release. Existing credentials generated with that model have to be re-generated before the 2021 Q3 release with one of the supported models to continue being functional. Keep in mind that after the 2021 Q3 release, existing credentials generated with that model won't work anymore. Remember that there is available a new endpoint to check the model used for creating the credentials /models/metadata/from-credential.

2 Introduction

Face biometrics is a state-of-the-art technology allowing to validate a person identity by means of his face. VERIDAS solution, das-Face, captures the unique features and characteristics of a face, generating a face biometric vector describing uniquely the person.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

The facial biometric vector is a mathematical descriptor obtained from the features found in a an image recording a person face. The conversion from face into a biometric vector is technically irreversible. Therefore, it is not possible to recover a person's face from the resulting biometric vector

The solution developed by VERIDAS and implemented in das-Face product is a service-oriented architecture that can be consumed via APIs.

Veridas facial biometric technology submission to the NIST (National Institute of Standards and Technology, USA) ongoing FRVT 1:1 verification 2020 ranking on top 40% worldwide in the category WILD, on the evaluation published on 27th July, 2020. VERIDAS achieved a False Negative Rate of 2.91% for a False Positive Rate of 0.01%, being VERIDAS system 0.2 points from the top-3.[a] Results shown from NIST do not constitute an endorsement of any particular system, product, service, or company by NIST.[b]

[a] NIST, *FRVT Report at 2020-07-27*.
[b] NIST, *FRVT on-going challenge*.

das-Face computes the similarity between faces recorded in images among other operations. The main operations implemented in this product are:

- Verification: This operation compares two faces to say if both are similar or not, so they can be considered from the same person.
- Liveness: This operation computes how likely a given capture contains a real face, or it is the result of a spoof attempt.
- Credential generation: This operations returns a biometric vector computed for the face represented in an image and returns it encoded and ciphered into a proprietary biometric credential format.

Veridas active liveness detection implemented in Selfie-Alive Pro was tested by iBeta to the ISO 30107-3 Biometric Presentation Attack Detection Standard and was found to be in compliance with Level 1.[a]

[a] iBeta, *Veridas PAD Level 1 Confirmation Letter*.

Beside to previous operations, das-Face supports operations over collections, i.e., galleries, of faces via the integration of das-Face API with the VERIDAS product called das-FaceBond. The operations offered by means of das-FaceBond are:

- Identification: das-Face offers its API to das-FaceBond for implementation one-face searches versus a collection of N faces, i.e., identification or 1:N search.
- Clustering: das-Face offers its API to das-FaceBond for clustering of collections of faces, i.e., making groups of faces with high similarity, helping to find duplicates in databases.

das-Face is offered as a REST API. The most common process to obtain the similarity between two faces is:

1. Two images are sent to the API.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

2. Both images are pre-processed, detecting the faces location in both images, and normalizing the detected faces in the way required for the next process.
3. The normalized face image is converted into an irreversible mathematical descriptor (facial biometric vector).
4. Both vectors are compared and a matching score between 0 and 1 is provided. The matching score represents a metric of the similarity between both face images. The higher the score, the greater the certainty to be the same person.
5. You can use this matching score to validate the identity of a customer. It is recommended to define a threshold within required confidence level using the FPR (False Positive Rate) and FNR (False Negative Rate) expected ratios. The calibration curves are shown at the document *das-Face Performance Report*.

VERIDAS does not store any personal data during the operation of this product. All the user information (i.e., both the images as well as the processed data) is immediately deleted.

3 Main Features

The main features of the das-Face facial biometrics engine are:

- Top-level technology: VERIDAS achieved top 40% wordwide in NIST FRVT 1:1.[1]
- Minimum resolution: Allows processing images with faces with at least 80px inter-eye distance.
- Verification time: around 1 second when comparing the biometric vector contained in a credential versus the face contained in an image.
- Face liveness detection: given an image input it is possible to detect if the recorded face is authentic or it is a spoof sample. This solution is in compliance to ISO 30107-3 PAD level 1, as tested by iBeta.

To ensure optimal performance, recommendation for images to be captured under specific conditions is needed. VERIDAS offers different proprietary SDKs for image photo and video recording, and they are available for different platforms (iOS, Android, HTML). These SDKs ensure that capture process is performed following the best conditions. Most relevant conditions are:

- Supported image formats: JPEG/JFIF, JPEG/EXIF, JPEG Blob, PNG, and TIFF.
- Supported video formats: MP4, WEBM, Quicktime.
- Images must be kept as returned by the SDK, any additional compression may lead to accuracy problems.
- Videos must be kept as returned by the SDK, any additional compression may lead to accuracy problems.

[1] NIST, *FRVT on-going challenge*.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

- Face must be of at least 80px between eyes to ensure optimum verification and biometric credential computations.
- Face must be of 150px width to ensure it can be processed by the face liveness detection systems. To increase accuracy, we recommended faces with more than 320px width.
- Face is expected to be frontal with the camera in the photograph. Less than 30 degrees in face pose axis (roll, pitch, yaw) is the optimum.
- Face appearance must be with no significant fish-eye distortion.

This API enables generation of biometric credentials, which are binary strings containing a biometric vector (embedding). Such vectors are a representation of the face used by das-Face for similarity comparison. Credentials have the following specifications:

- Encoding: das-Face returns credentials encoded in base64. It is recommended to decode the string to binary before writing it to a persistent storage. Any POST request containing a credential must encoded it in base64 to be understood by das-Face.
- Length:
  - 2896 bits are required for credentials in binary format generated by /v2/credential/photo.
  - The model 20210203 requires 9040 bits in binary format.

4 System accuracy report

The accuracy of the das-Face system is given in the document *das-Face Performance Report*.

5 Use cases

This section describes the main use cases of the das-Face API. Take into account that these are generic concepts that can be composed or orchestrated in a more complete pipeline to perform processes like customer on-boarding (vali-Das product by VERIDAS).

5.1 Enrollment

das-Face is an stateless service and then it did not implement a full enrollment use case. Usually enrollment is not required because das-Face API may receive two photographs (or a photograph and a video) and perform the identity verification between the two faces located in both images. The customer is responsible of storing one of the images for enrollment if that is enough for the operation of their systems.

However, enrollment may be necessary and in such a use case das-Face allows creating a biometric credential, and returning the particular metadata of the model used for the biometric operation. The customer should persist in the database:

- the enrollment image,
- the biometric credential, and
- the biometric model metadata.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Credentials must be decoded from base64 into a binary representation before persisting them in any device, and it must be encoded again to base64 in any following communication with das-Face. The image and the model information are required to regenerate biometric credentials when das-Face is updated with improved biometric models.

So, for enrollment, das-Face computes a proprietary irreversible mathematical representation of a face, called biometric credential, plus some metadata information required to identify the biometric model. The biometric credential is signed and encrypted using a key that is different for each of our customers, making them non-interoperable between customers.

The credential can be generated by calling to POST $/\mathrm{v2/credential/photo}$ endpoint, which will generate it with the biometric model 20200514. On operation, the generated credential can be compared with the face located in a given photograph.

Warning!!! When the biometric model is updated, a notice will be given in this documentation, and the customer will be given a time window for plan an update their credentials database.

Notice that credential generation is optional and may require a specific agreement.

5.2 Verification

This use case of das-Face allows to compute the similarity between the faces shown in two photographs, or between a video and a face photograph. To do so, the client must perform a request POST $/\mathrm{v2/verification/photo}$ for photo comparison, or to POST $/\mathrm{v2/verification/video}$ for photo and video comparison. In both cases, the system response is a JSON with a $\mathrm{confidence}$ field, indicating a value in range $[0.0, 1.0]$.

If the customer has created biometric credentials, as explained in the 5.1 Section, it is possible to perform verification of a photo and a credential by requesting POST $/\mathrm{v2/verification/credential}$ endpoint.

In all these cases the system can be configured to perform the operation using $\mathrm{selfie\text{-}mode}$ or $\mathrm{document\text{-}mode}$. Selfie-mode and document-mode differ in how the system response is calibrated, and hence, both modes follow different calibration curves that are depicted in the *das-Face Performance Report*. The $\mathrm{selfie\text{-}mode}$ option is calibrated to work with selfie-like images, and the $\mathrm{document\text{-}mode}$ option is calibrated to compare a selfie versus a face crop extracted from a photograph of an ID card.

The $\mathrm{document\text{-}mode}$ is convenient for vali-Das on-boarding product. In such situation, das-Face expects as input the face crop returned by vali-Das and a selfie photograph. das-Face will look for a face if a whole ID card photograph is given, but the success of the procedure is dependent on the specific ID card design.

5.3 Passive liveness detection

Liveness detection allows to compute the confidence of a live capture being performed over a bonafide person, or otherwise, if it belongs to a spoof sample build by a fraudster. das-Face implements passive and active liveness procedures.

This passive procedure detects user liveness by looking into a selfie capture. das-Face will process the given photograph and return a confidence number in range $[0.0, 1.0]$ which is calibrated with curve given in the *das-Face Performance Report*. To operate this functionality, the client must perform a request POST $/\mathrm{v2/authenticity/photo}$.

Notice that photo authenticity is optional and may require a specific agreement.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

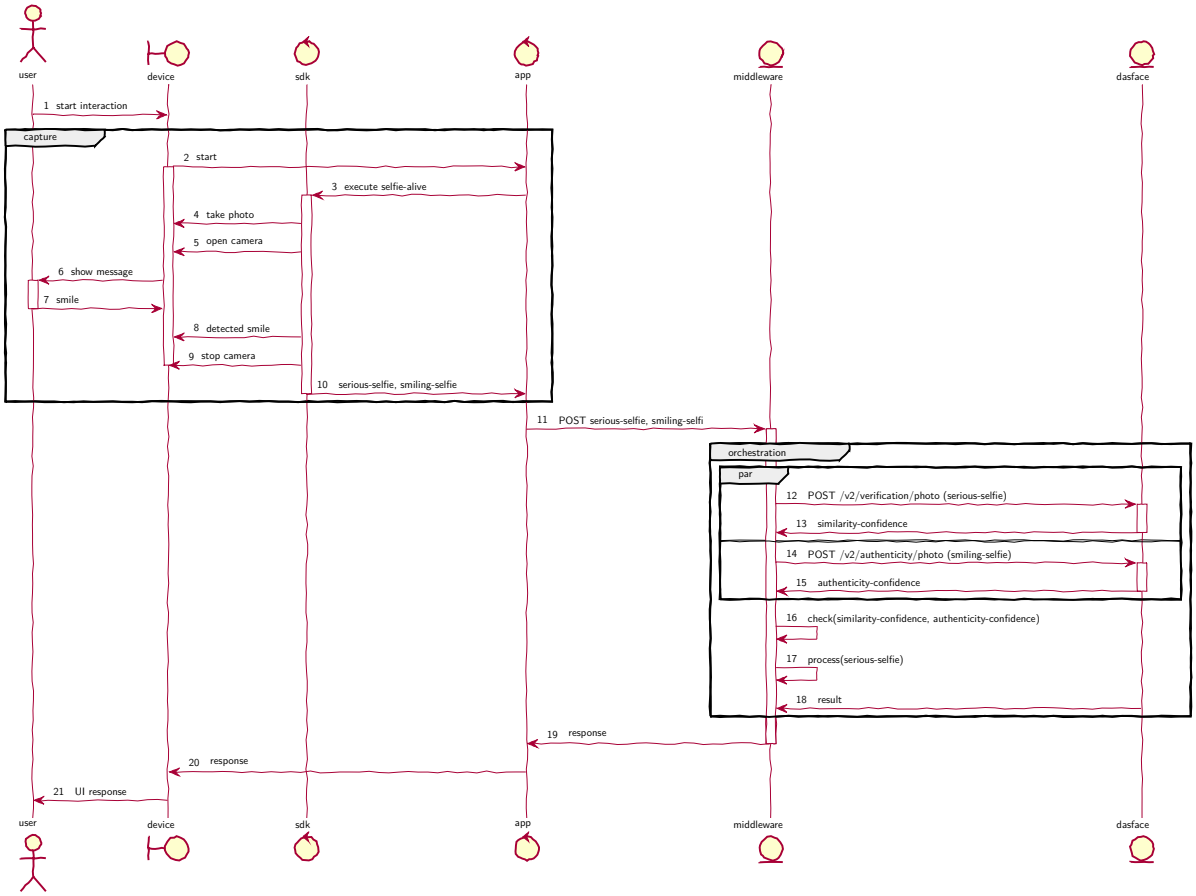

Figure 1: Interaction for selfie-alive with smile for liveness check.

5.4 Selfie-Alive liveness detection

This is a use-case of active liveness detection that is orchestrated by means of VERIDAS SDKs. On das-Face side, the liveness detection is performed passively by analizing a photo capture, but the end-to-end system consists in asking the user to smile in front of the device camera, and send the evidences in the proper order to the proper endpoints in das-Face.

The procedure depicted in the interaction diagram shown at Figure 1 is a general with the purpose of checking the face liveness using selfie-alive SDK. Liveness is verified by the procedure 'check()', which basically compares that $\text{similarity-confidence} > \text{th1}$ and $\text{authenticity-confidence} > \text{th2}$, where usually th1 is a high threshold (0.85, 0.90, ...), and th2 must be over 0.5 (0.5, 0.6, 0.7, ...). The actual thresholds depend on the trade-off between FPR and FNR required by the customer application, see Tables in *das-Face Performance Report*.

Once the liveness is verified by $\text{check}()$, the middleware can do one of these things in the $\text{process}()$ function:

- Enrolling the serious-selfie for future authentication.
- Compute a biometric credential via das-Face API, and enroll the credential for future authentication.

  Notice that credential generation is optional and may require a specific agreement.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

- Authenticate an user via das-Face verification API, using the new captured serious-selfie with an enrollment retrieved from the customer database.

To ensure maximum performance, all calls to das-Face can be performed in parallel, processing all the scores accordingly when they are all received.

5.5 Selfie-Alive Pro liveness detection

This use-case is based on a challenge-response design. das-Face will generate a challenge that should be reproduced using native VERIDAS SDKs for iOS and Android. Then, the captured evidences will be sent to das-Face for its final analysis.

Veridas active liveness detection implemented in Selfie-Alive Pro was tested by iBeta to the ISO 30107-3 Biometric Presentation Attack Detection Standard and was found to be in compliance with Level 1.[a]

[a] https://www.ibeta.com/wp-content/uploads/2020/12/201215-Veridas-PAD-Level-1-Confirmation-Letter.pdf

Selfie-Alive Pro functionality is a form of face liveness detection implemented following a challenge-response schema. The Figure 2 depicts the whole interaction with final user, app, sdk and das-Face. In a first stage, the client must operate a request POST /v2/challenges/generation/sequential to generate a new challenge token. The response of this request is a JWS string with application/jose mime type. In a second stage, the client needs to record an interaction with a user face, following the indications of the challenge. This second stage is implemented completely by Veridas SDKs. In a third stage, the client must request POST /v2/challenges/analysis/video-photo with all the evidences returned by the SDK (among them, a video recording of the face). The system will reply with a confidence metric in range $[0.0, 1.0]$, calibrated as indicated in the curve of Selfie-Alive Pro in the *das-Face Performance Report*.

Notice that Selfie-Alive Pro is optional and may require a specific agreement.

6 API v2

The API described in this appendix has the following attributes:

- This API enables generation of biometric credentials, which are binary strings containing a biometric vector (embedding). Such vectors are a representation of the face used by das-Face for similarity comparison. Credentials have the following specifications:
  - Encoding: das-Face returns credentials encoded in base64. It is recommended to decode the string to binary before writing it to a persistent storage. Any POST request containing a credential must encoded it in base64 to be understood by das-Face.
  - Length: depends on the biometric model, please, look below for the model specs.
- Available biometric models:
  - * Model tag: 20210203
    * Model hash: bab0b604e3cb088dd6c484a9774505f5545d3e5837874f3ec0172af5
  - * Model tag: 20200514
    * Model hash: 904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

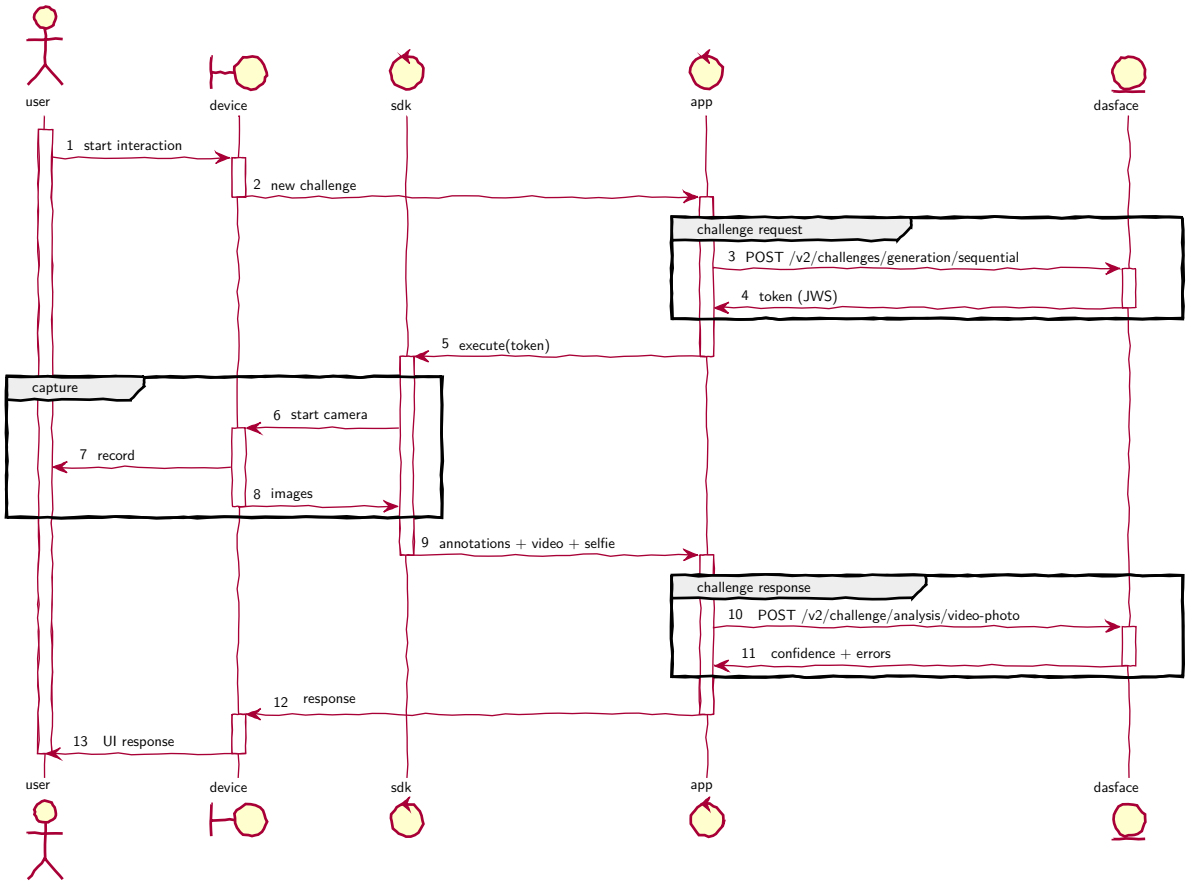

Figure 2: Interaction for Selfie-Alive Pro,based on challenge-response video liveness procedure.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

  - * Model tag: 20190813
    * Model hash: 3a9e9d5ffd5de4c212c2aff26eeca523fb69754e604894520b32e4ed

- All endpoints in this API v2 are prefixed with /v2/.
- The server processes requests in application/json and multipart/form-data media types.
  - In case of application/json, all files should be encoded in base64 and sent as a string field in the JSON document.
  - For multipart/form-data, files will be sent as different parts of the body, using binary encoding formats (usually as media/jpeg, media/png, ..., depending on the media file type).
- Responses are always in application/json media type.
  - Successful responses (HTTP status codes 2XX and 3XX) are described in this documentation by specifiying the returned JSON fields. For each JSON field, the following properties are described:
    * Name: The name of the field in the JSON object.
    * Req.: Indicates if this field is required or not. When it is required, the server will produce an error when the indicated field is not found in the request.
    * Type: The expected type of the field content. It could be: file, string, integer, number, array.
    * Description: A paragraph describing the purpose and the content of the field.
  - Responses for request errors and server exceptions (HTTP status codes 4XX and 5XX) always follow the same JSON structure:

```
{
  "code": "FaceNotFoundError",
  "message": "Unable to locate a face in the image"
}

```

    Then, they are documented by specifying the available code names and a message example. For different convenient reasons, the server may return more fields in such a JSON object, and the client must ignore them.
  - Finally, bad formed requests, i.e., with from validation errors, will produce a 400 response with "code": "FormValidationError". This particular error code adds to previous JSON structure the optional field "errors", which is an array of 2-tuples, where each tuple indicates the field name and a message of the failure reason. The following example illustrates one of these cases:

```
{
  "code": "FormValidationError",
  "message": "Incorrect parameters",
  "errors": [
    [
      "anchorImage",
      "This field is required"
    ]
  ]
```


All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

```
10   }
11
```

Similarly to previous case, due to convenience, the server may return more fields in the JSON object, and the client must ignore them.

The following are the average response times of the main endpoints:

Endpoint	Average (ms)
/v2/authenticity/photo	1180
/v2/authenticity/photo/unequal-pair	1290
/v2/credential/photo	910
/v2/verification/credential	910
/v2/verification/photo	1820
/v2/verification/video	2430
/v2/challenges/analysis/video-photo	5280

6.1 *GET* /dasface/v2/alive

The service receives a GET request with no params, and returns a 204 status code indicating that the server is up.

Aliases for this path:

- /alive

Response: 204

Empty response.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.2 *POST* /dasface/v2/authenticity/photo

Computes the authenticity confidence of a selfie photo. The service receives a POST request containing a selfie image and it the confidence of the system about the authenticity of the photo. This means, the closer the value to 0, the most likely it is a spoofing attempt. This endpoint is optional and may require a specific agreement.

Aliases for this path:

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

- /authenticity/photo

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for photo authenticity.

Name	Req.	Type	Description
targetImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.

Response: 200

Returns the confidence of the given image to be authentic, that is, the closer to 0, the most likely it is a spoofing attempt.

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FaceTooSmallForIAS	Image size is too small for analyzing its authenticity
FormValidationError	Unable to validate form fields

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.3 *POST* /dasface/v2/authenticity/photo/unequal-pair

Computes confidence that two images are not a capture of same photo (duplicate attack). The service receives a POST request with two face images and returns the confidence number which represents how likely both images are not a capture of the same photo (duplicate attack). A duplicate attack consists of sending two images containing the same photo, for instance, one image could be a photograph for visa and the other a photo of a passport issued using previous photo, therefore, both images contain the same photo.

Aliases for this path:

- /authenticity/photo/unequal-pair

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for unequal pair of photos.

Name	Req.	Type	Description
anchorImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.
targetImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.

Response: 200

Returns the confidence of both images being different, that is, the closer to 0, the more likely it is a duplicate attack; the closer to 1, the most likely it is not.

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FaceTooSmallForIAS	Image size is too small for analyzing its authenticity
FormValidationError	Unable to validate form fields

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.4 *POST* /dasface/v2/challenges/analysis/video-photo

Performs face verification of a selfie and a video recording, and authenticates both, the video recording and selfie. The video recording is returned by SDK and sent to dasFace as a file. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /challenges/analysis/video-photo

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for authenticity operation based on video and selfie.

Name	Req.	Type	Description
anchorImage	yes	file	A face photo to be compared with the face shown in the video. It must be given as a file. When body is application/json, the field must contain the file content encoded in base64.
annotations	yes	file	File with annotations of video in WebVTT format. It should contain the result given by the SDK. It must be given as a file. When body is application/json, the field must contain the file content encoded in base64.
targetVideo	yes	file	Video recording of the challenge response. It must be given as a file. When body is application/json, the field must contain the file content encoded in base64.
token	yes	file	JWS token as returned by a challenge request to /api/v2/challenges/sequential. It can be given as a file or as a string variable.

Response: 200

The response containing authenticity and similarity confidences for all the given media resources. The returned confidence can be a number in [0.0, 1.0] or a None. The None value indicates a sort of problem during the operation, and in these cases, the errors response field will contain details about the failure.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].
errors	yes	array	List of errors produced while computing the returned confidence. Each error is reported as a 2-tuple with elements (code, message), where the code is an error code name, and the message is a text explaining the problem. The error code names are: FaceNotFoundError, FaceAlignmentError, MoreThanOneFaceError, NotEnoughVideoDataError, TooShortVideoError, and FaceTooSmallForIAS.

Response: 400

Request format error.

Content-Type: application/json

code	message
FormValidationError	Unable to validate form fields
ExpiredOrInvalidChallengeError	Given challenge is expired or invalid

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

6.5 *POST* /dasface/v2/challenges/generation/sequential

Generates a random sequential challenge, returning a JWS token (RFC7515) containing a signed JSON with a challenge description. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /challenges/generation/sequential

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request a random challenge.

Name	Req.	Type	Description
expiration	no	integer	Expiration time in seconds, by default it is 1800 seconds.
length	no	integer	Desired length of the challenge, by default it is 3. The minimum is 1 and the maximum is 10. The recommended values are 3 for standard security, and 6 for high security.

Response: 200

Returns a JWS token containing the challenge description. The challenge is a JSON object.

Content-Type: application/jose

Response: 400

Request format error.

Content-Type: application/json

code	message
FormValidationError	Unable to validate form fields

Response: 415

Incorrect request media type.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.6 *POST* /dasface/v2/credential/photo

This endpoint is used to generate a biometric credential from a given image, using the latest available model. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /credential/photo

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for biometric credential generation.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Name	Req.	Type	Description
mode	no	string	Configures the operation mode of the biometric model. It can be: ▪ selfie-mode: To configure the system for comparison of two selfie photos. ▪ document-mode: To configure the system for comparison of a selfie photo versus a crop of the face printed in an ID card. By default it is selfie-mode.
targetImage	yes	file	As multipart/form-data, it should be a file, as application/json, the file content encoded in base64.

Response: 200

Returns one biometric credential for the given image file.

Content-Type: application/json

Name	Req.	Type	Description
credential	yes	string	A biometric credential string encoded in base64.
model	yes	object	An object with two items: *hash* of the model used to generate the biometric credential, and the operation *mode* used to configure the model.

Example

```
{
    "credential": "ABC6274F",
    "model": {
        "hash": "b0bf475e5344e816f12b83c13c075a3256ef95e60ac1cdc273aef59f",
        "mode": "selfie-mode"
    }
}
```

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FormValidationError	Unable to validate form fields
MetadataValidationError	Incorrect metadata in given credential.

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.7 *GET* /dasface/v2/models

Get the available embedding models and their operation modes. The service receives a GET request and returns a list of all available models and operation modes. Models are identified by a hash number, and they are given a numeric tag. The numeric tag is always a date string in the form YYYYMMDD. The hash is an hexadecimal number represented in a string value. Operation modes are given as a string with words separated by dashes, e.g., selfie-mode. This request also returns a "methods" field, reserved for future uses. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /models

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Response: 200

Returns list of models sorted by their numeric tag (descending order).

Content-Type: application/json

Name	Req.	Type	Description
models	yes	array	An array with all available models and modes, for each combination, it is indicated: tag, hash, mode, and methods. See the example for more details.

Example

```
[
    {
        "tag": "20200514",
        "hash": "904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15",
        "mode": "selfie-mode",
        "methods": [
            "Face"
        ]
    },
    {
        "tag": "20200514",
        "hash": "904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15",
        "mode": "document-mode",
        "methods": [
            "Face"
        ]
    }
]
```

Response: 500

Server error response.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.8 *POST* /dasface/v2/models/metadata/from-credential

This endpoint retrieves the operational model used to generate an existing biometric credential. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /models/metadata/from-credential

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for biometric credential metadata.

Name	Req.	Type	Description
credential	yes	string	A biometric credential as previously generated from an image. It is an string containing the biometric credential as returned by the service, that is, in base64 encoding.

Response: 200

Returns JSON array containing data about operational model used to generate each of the given input biometric credentials

Content-Type: application/json

Name	Req.	Type	Description
metadata	yes	object	An object with three items: *hash* of the model used to generate the biometric credential, the operation *mode* used to configure the model, and the latest release *tag* containing such a model.

Example

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

```
{
    "metadata": {
        "hash": "b0bf475e5344e816f12b83c13c075a3256ef95e60ac1cdc273aef59f",
        "mode": "document-mode",
        "tag": "20200514"
    }
}
```

Response: 400

Request format error.

Content-Type: applciation/json

code	message
ObsoleteCredentialModelError.	The model used to generate the given credential is obsolete.
FormValidationError	Unable to validate form fields

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.9 *POST* /dasface/v2/models/{hash}/{mode}/credential/photo

This endpoint is used to indicate the system which operational model must be used to generate a biometric credential. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /models/{hash}/{mode}/credential/photo

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		
hash	path	A hash given as hex digest, same as returned by {/v2/models}.		
mode	path	A mode given as selfie-mode, document-mode.		document-mode

Request Body

Request for a biometric credential generation with parameters.

Name	Req.	Type	Description
targetImage	yes	file	As multipart/form-data, it should be a file, as application/json, the file content encoded in base64.

Response: 200

Returns one biometric credential for the given image file.

Content-Type: application/json

Name	Req.	Type	Description
credential	yes	string	A biometric credential string encoded in base64.
model	yes	object	An object with two items: *hash* of the model used to generate the biometric credential, and the operation *mode* used to configure the model.

Example

```
{
    "credential": "ABC6274F",
    "model": {
        "hash": "b0bf475e5344e816f12b83c13c075a3256ef95e60ac1cdc273aef59f",
        "mode": "selfie-mode"
    }
}
```

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FormValidationError	Unable to validate form fields
MetadataValidationError	Incorrect metadata in given credential.

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.10 *POST* /dasface/v2/verification/credential

Computes the face similarity between a face image and a previously generated biometric credential, and returns how similar are both faces. This endpoint is optional and may require a specific agreement.

Aliases for this path:

- /verification/credential

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for face verification with a biometric credential.

Name	Req.	Type	Description
anchorImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.
targetCredential	yes	string	A biometric credential as previously generated from an image. It is an string containing the biometric credential as returned by the service, that is, in base64 encoding.

Response: 200

Returns the confidence (or similarity) between both faces, being more similar as much close this number is to one. The number is in range [0,1].

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FormValidationError	Unable to validate form fields
FormatNumberError	Incorrect format number in credential.
CorruptedSecretError	Unable to decode the credential with known secret pass.
MetadataEqError	Incompatible metadata in given credentials.
MetadataValidationError	Incorrect metadata in given credential.

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.11 *POST* /dasface/v2/verification/photo

Computes the face similarity between two face images. The service receives a POST request with two image files containing one face per image, and returns how similar are both faces.

Aliases for this path:

- /verification/photo

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for face verification.

Name	Req.	Type	Description
anchorImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.
mode	no	string	Configures the operation mode of the biometric model. It can be: ▪ selfie-mode: To configure the system for comparison of two selfie photos. ▪ document-mode: To configure the system for comparison of a selfie photo versus a crop of the face printed in an ID card. By default it is selfie-mode.
targetImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.

Response: 200

Returns the confidence (or similarity) between both faces, being more similar as much close this number is to one. The number is in range [0,1].

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Response: 400

Request format error.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
FormValidationError	Unable to validate form fields

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

6.12 *POST* /dasface/v2/verification/video

Computes similarity between a face photo and a face video. The service receives a POST request with a video and a face photo, and returns the similarity between the face on the video and the face on the image.

Aliases for this path:

- /verification/video

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Parameters

Name	In	Description	Default	Example
X-Veridas-RTag	header	This header allows to tag this query with a personal string. This tag needs to be a ASCII string with a maximum size of 64 characters.		

Request Body

Request for video face verification.

Name	Req.	Type	Description
anchorImage	yes	file	An image containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.
targetVideo	yes	file	A video containing the face of a person. When body is multipart/form-data, this field should be a file. When body is application/json, the field must contain the file content encoded in base64.

Response: 200

Returns the confidence of both faces being different, that is, the closer to 1, the most likely the face in the video is the same in the image.

Content-Type: application/json

Name	Req.	Type	Description
confidence	yes	number	A probability number in range [0,1].

Response: 400

Request format error.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

Content-Type: application/json

code	message
FaceNotFoundError	Unable to locate face in given image
FaceAlignmentError	Unable to align face in given image
MoreThanOneFaceError	More than one face located at given image
ZeroLengthVideoError	An empty video was received
VideoExtractionError	Unable to decode/extract video data
InvalidFPSVideoError	Retrieved FPS number is invalid
InvalidNumFramesVideoError	Retrieved number of frames is invalid
FormValidationError	Unable to validate form fields

Response: 415

Incorrect request media type.

Content-Type: application/json

code	message
UnsupportedMediaTypeError	Request media type is not supported by the server.

Response: 500

Server error response.

Content-Type: application/json

code	message
ServerError	Internal Server Error

7 References

[1] iBeta. *Veridas PAD Level 1 Confirmation Letter*. Dec. 2020. URL: https://www.ibeta.com/wp-content/uploads/2020/12/201215-Veridas-PAD-Level-1-Confirmation-Letter.pdf.

[2] NIST. *FRVT on-going challenge*. 2020. URL: https://www.nist.gov/programs-projects/face-recognition-vendor-test-frvt-ongoing.

[3] NIST. *FRVT Report at 2020-07-27*. 2020. URL: https://github.com/usnistgov/frvt/raw/nist-pages/reports/11/frvt_11_report_2020_07_27.pdf.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

A Changelog

A.1 das-Face v3.0 (2021-Q1)

This version comes with two major changes related to biometric models and antispoofing techniques.

- New biometric model which reduces false positive and false negative rates, and improves the accuracy on different demographic cases (Caucasian, African, Indian, Asian). The model improves the accuracy of selfie-mode from 99.8% to 99.9% and the funnel of document-mode is improved by two points maintaining the same security level, obtaining TPR=2.43% @ FPR=0.62% at 0.7 threshold. The new model is identified by:
  - Model hash: 904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15
  - Model tag: 20210203
- Because model 20210203 generates larger biometric credentials than earlier models, the default endpoint /credential/photo will generate them using model 20200514, so by default credentials have the 2896 bits length as usual.
- When security is a must, and having a use case that could afford using larger biometric credentials, the model 20210203 is available at the endpoint:
  - /models/<hash>/<mode>/credential/photo.

  The biometric credentials for model 20210203 are of 9040 bits length.
- An update of the Selfie-Alive Pro (SAP) use case (/challenges/analysis/video-photo) decreasing the time of the operation in a 26%. The system is calibrated at threshold 0.7 for BPCER=2.7% and APCER=0.9%.
- An update of the passive AS solution (/authenticity/photo) calibrated at threshold 0.7 for BPCER=3.2% and APCER=7.8%, reducing a 50% the APCER of the previous das-Face version. The system requires additional 300ms per query (a total of 1185ms per query).
- **Deprecation announcement**: The biometric model with:
  - model hash: 3a9e9d5ffd5de4c212c2aff26eeca523fb69754e604894520b32e4ed
  - model tag: 20190813

  is marked as deprecated, enforcing deprecation for the **2021 Q3** release. Existing credentials generated with that model have to be re-generated before the 2021 Q3 release with one of the supported models to continue being functional. Keep in mind that after the 2021 Q3 release, existing credentials generated with that model won't work anymore. Remember that there is available a new endpoint to check the model used for creating the credentials /models/metadata/from-credential.

A.2 das-Face v2.7.2 (2021-Q1)

- Fixes low scores at '/verification/video' when using mode=document-mode.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

A.3 das-Face v2.9 (2021-Q1)

- Add support for BBVA SEMaaS traceId field (using the x-rho-traceid header).

A.4 das-Face v2.8 (2021-Q1)

- Added support for JPEG blobs.

A.5 das-Face v2.7 (2020-Q4)

This new version of das-Face comes with new liveness detection capabilities.

- Improved passive liveness detection with the following figures:
  - Improves replay-attacks APCER from 27% in das-Face v2.5.2 to 5.3% in das-Fave v2.6.7.
  - Achieves BPCER=11.7%
  - These operational figures are valid for selfie-alive and vali-Das on-boarding products.
- Adds Selfie-Alive Pro use case, VERIDAS liveness detection using an active procedure.
  - The system performs with APCER=0.1% and BPCER=12.9%.
  - Supports videos recording a sequence of random head movements.
  - This solution is in compliance to ISO 30107-3 PAD level 1, as tested by iBeta.
- Support for a contextual data header: $\mathrm{X\text{-}Veridas\text{-}RTag}$ header is logged in every query allowing to build statistics and billing segregated by customer, use cases, sub-clients and so on.

Deprecation enforcement
Enforcement of deprecation for the model identified with:

- Tag: 20180827
- Hash: $\mathrm{b0bf475e5344e816f12b83c13c075a3256ef95e60ac1cdc273aef59f}$

Existing credentials generated with such model won't be compatible any more with das-Face.

A.6 das-Face v2.6 (never released)

This version was used internally to calibrate the liveness detection released in das-Face v2.7.

A.7 das-Face v2.5 (2020-mid-Q3)

- Fix bug in 'v2/verification/video' when more than one face was detected in the video.

A.8 das-Face v2.4 (2020-Q2)

- New biometric model which reduces false positive and false negative rates, and improves the accuracy on different demographic cases (Caucasian, African, Indian, Asian). The model improves accuracy of selfie-mode form 99.6% to 99.8% and document-mode from 97.87% to 98.34%. The new model is identified by:
  - Model hash: $\mathrm{904fa9ef6e71ef541f20a95d3dc97821b7af43b8cd2c1bb3eb09df15}$

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

  - Model tag: 20200514

- Added a new endpoint to retrieve the operational model from the biometric credential metadata, i.e., model hash and operation mode chosen to generate the given credential.
- Added a new endpoint to get the operational models active in das-Face.
- Added a description of the orchestration required for selfie-alive.
- **Deprecation announcement** The biometric model with:
  - model hash: b0bf475e5344e816f12b83c13c075a3256ef95e60ac1cdc273aef59f
  - model tag: 20180827

  is marked as deprecated, enforcing deprecation for 2020 Q4 release. Existing credentials generated with that model have to be re-generated before 2020 Q4 release with one of the supported models to continue being functional. Keep in mind that after 2020 Q4 release, existing credentials generated with that model won't work anymore. Remember that from this release there is available a new endpoint to check the model used for creating the credentials.

A.9 das-Face v2.3 (2020-Q1)

- The endpoint POST /v2/authenticity/photo (antispoofing) accepts images images taken with Veridas HTML SDKs.
- **Deprecation reminder**: API v1 endpoints are not supported any more.

A.10 das-Face v2.2 (2019-Q4)

- Improved log messages, increasing their exploitability.
- Recall deprecation of API v1 endpoints, it will be enforced at 2020 Q1.

A.11 das-Face v2.1 (2019-Q3)

- Improved document-mode biometric model for selfie-vs-document functionality.
- Improved selfie-mode biometric model for selfie-vs-selfie functionality.
- Improved anti-spoofing detection for photo authenticity endpoint, increasing accuracy from 90.3% to 92.2% for replay-attacks.
- Replacement of dataset used to compute photo authenticity performance curve (anti-spoofing). The new replay-attack dataset is larger, uses more devices and has been carried out under more complex conditions.
- Notice that due to the new biometric model for document-mode and selfie-mode, all endpoints are expected to use the new model.
- Recall the deprecation of API v1 endpoints. We encourage our customers to start using the new API v2 because API v1 will be removed at 2020 Q1.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

A.12 das-Face v2.0 (2019-Q2)

- New API v2 with better semantics, syntax and improved error handling.
- New endpoint /v2/credential/photo to generate a biometric credential from an image. The returned credential contains the biometric data of the face found in the image.
- New endpoint /v2/verification/credential to compare the image of a face with a previously generated biometric credential.
- Deprecation of API v1 endpoints. We encourage migration to API v2 as soon as possible.

B License

The following clauses set the terms, rights, restrictions and obligations on using this Software, created and owned by VERIDAS DIGITAL AUTHENTICATION SOLUTIONS, S.L. ("VERIDAS" or the "Licensor"), without prejudice to the provisions laid down in the contracts subscribed by the Licensor and your entity (the Licensee), which shall prevail over this file.

B.1 LICENSE GRANT

VERIDAS hereby grants to the Licensee a non-exclusive, non-assignable and non-transferable, non-commercial, indivisible, without the rights to create derivative works license for the term specified in the Offer to use the offered software (the "Software") for the specific purpose specified between the Parties and/or in the Offer, subject to the terms and conditions contained herein and other legal restrictions set forth in third party software used while running the Software.

The Software has different components that can be used for several applications. However, the license is granted over the Software licensed components as a whole, and no separated use is permitted other than the specific purposes agreed by the Licensor and the Licensee.

The Software is comprised of proprietary code. However, the Software may include certain third-party components with separate legal notices or governed by other agreements, as may be described in the Software. Even if such components are governed by other agreements, the disclaimers and the limitations on and exclusions of damages below also apply.

On specific products, if necessary, VERIDAS may install a computer application, in some cases connected with an external server, that allows VERIDAS to verify that the system is updated and payments are correctly made.

B.2 USE OF THE SOFTWARE

2.1. The Licensee cannot use the Software for other purposes than as specified in the Offer.

2.2. The Licensee may permit its employees to use the Software for the purposes agreed by the parties and/or described in the Offer, provided that the Licensee takes all necessary steps and imposes the necessary conditions to ensure that all employees using the Software do not commercialize or disclose the contents of it to any third party, or use it other than in accordance with the terms herein.

2.3. The Licensee will not distribute, sell, license or sub-license, lease, trade or expose for sale the Software to a third party.

2.4. No copies of the Software are to be made other than as expressly approved by VERIDAS.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

2.5. No changes to the Software or its content may be made by Licensee.

2.6. The Licensee will provide technological and security measures to ensure that the Software, which the Licensee is responsible for, is physically and electronically secure from unauthorized use or access.

2.7. The Licensee shall ensure that the Software retains all VERIDAS copyright notices and other proprietary legends and all trademarks or services marks of VERIDAS, as specified in clause 3 below.

2.8. The Licensee shall not, under any circumstances, use reverse engineering practices on the Software.

2.9. The Licensee is responsible for extending the obligations herein to Clients, to the extent they may apply.

B.3 INTELLECTUAL PROPERTY RIGHTS

3.1. Intellectual Property Rights means all rights in and to any copyright, trademark, trading name, design, patent, know-how, trade secrets and all other rights resulting from intellectual activity in the industrial, scientific, literary or artistic field and any application or right to apply for registration of any of these rights and any right to protect or enforce any of these rights.

3.2. All Intellectual Property Rights over and in respect of the Software are owned by VERIDAS. The Licensee does not acquire any rights of ownership in the Software, and it must use the Intellectual Property Rights exclusively as required for reasonable and customary use within the purposes of the License.

3.3. Any modification made on the Software in order to adapt it for the provision of the service to the Licensee and/or the Client, shall be include in the Intellectual Property Rights as defined in clause 3.2 above.

B.4 LIMITATION OF LIABILITY

4.1. To the extent permitted under the law, the Software is provided under an "AS IS" basis. The Licensee acknowledges and agrees that neither VERIDAS nor its board members, employees or agents, will be liable for any lost or damage arising out of or resulting from VERIDAS' provision of the Software under this License, or any use of the Software by the Licensee, the Client or their employees. The Licensee hereby releases VERIDAS to the fullest extent from any such liability, loss, damage or claim, both its own or of the Clients. Regarding the processing of personal data with the Software, the Licensee acknowledges and agrees that the Licensor is no liable for the data collected by the use of the Software within the scope of the Licensee's activities.

This limitation shall apply to any issue related to the software, services, contents (including code) found at third-party websites or third-party programs.

4.2. The Licensee must indemnify, defend and hold harmless the Licensor, its board members, employees and agents from and against any and all claims (including third party claims), demands, actions, suits, expenses (including attorney's fees) and damages (including indirect or consequential loss) resulting in any way from: Licensee's and Licensee's employee's use or reliance on the Software; any breach of the terms of the License by the Licensee or its employees; any other act of Licensee that can be considered negligent.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.

4.3. Notwithstanding the previous general clause, VERIDAS expressly excludes any liability resulting from:

(a) willful, fraudulent, deliberately unlawful acts, penalized as a criminal offence, or which are voluntarily against the law, carried out by or against the Licensee or a Client, or their employees;

(b) any fact or circumstance, real or suspected, the Licensee or the Clients know about or could have reasonably foreseen, and that may affect, in any way, to the correct functionality of the Software;

(c) mechanical fails, electric fails (including interruptions, outages, overvoltages or power cuts) and fails on telecommunication or satellite transmission systems, given that those fails are not due to an act or omission of VERIDAS or to an error of the delivered Software;

(d) damage or loss of the Licensee's or Client's data stored or hold in VERIDAS' systems, as well as the costs resulting from such circumstance; or

(e) any act or claim alleging, derived from or based on funds, money or value transfers or any other negotiable instrument for or from a bank or financial institution.

4.4. Licensee and Clients acknowledge that the Software has an associated error rate which makes not possible, considering the state of the art, to guarantee a complete level of reliability. In this sense, VERIDAS provides information about the levels and error rates of every Software version. On the other hand, the outputs of the system are not binary but they offer a probabilistic result that has to be configured by the Licensee or the Client.

The facial verification functionality of this Software has been tested against the LFW database, with results up to a 99.8% precision (FNR$= 0.4\%$ at FPR$\leq 0.4\%$). This level of reliability is subject to image capture conditions during the process, and therefore the Licensor shall not be responsible of any use the Licensee may make of the Software in different environments than those recommended by the Licensor.

As a result, it is the responsibility of the Licensee and/or the Client to carry out any necessary internal control evaluation, to implement due diligence measures in accordance with the regulation they have to comply with (including without limitation: regulation on the prevention of money laundering, citizen security, border controls, etc.) and to evaluate the convenience of the Software as an instrument for complying with current legislation.

VERIDAS provides some tools that may help the Licensee and/or the Client to implement prevention mechanisms, but it is not responsible for the study of the convenience of its implementation, the specific configuration of the Software, or the use of the Software (expressly including the validation results obtained with the Software).

4.5. Licensee is responsible for communicating and transferring the previous limitations of liability to the Clients.

All rights reserved — This document contains confidential information, property of Veridas Digital Authentication Solutions, S.L., and cannot be reproduced, copied, or revealed to third parties, without the express written authorization of Veridas. The information of this document must be kept secret and used in the exclusive benefit of Veridas.