



Guidelines on the self- assessment questionnaire PDAOFI

User Guide

Guidelines on the self-assessment questionnaire PDAOFI

User Guide

Update information

Date	Version	Changes
10/01/2025	1.0	Initial version

Table of CONTENTS

1. Transmission of the self-assessment questionnaire PDAOFI to the CSSF.....	5
1.1. Available channels	5
1.1.1. Application Programming Interface (API).....	5
1.1.2. eDesk	5
1.2. Technical format for API channel.....	6
1.2.1. Input files.....	6
1.2.2. CSSF feedback files	7
2. Contact information	9

1. Transmission of the self-assessment questionnaire PDAOFI to the CSSF

1.1. Available channels

The self-assessment questionnaire PDAOFI can be submitted via an online form available on the eDesk platform.

It is possible to pre-fill the questionnaire via the API system (S3) by using a structured exchange file.

1.1.1. Application Programming Interface (API)

The method of transmitting reports via Application Programming Interface can be found on our website at the following link: <https://www.cssf.lu/en/Document/methods-of-transmitting-reports-via-s3-application-programming-interface-technical-guidance/>

On the eDesk IT management console, the "IT Expert" must create a bucket "**SAQ PDAOFI**".

The "**SAQ PDAOFI**" bucket must be created by a "IT expert" linked to the entity concerned by the questionnaire ("**the PDAOFI**").

The API channel allows:

- the submission of a structured exchange file to pre-fill the questionnaire;
- the reception of corresponding feedback.

Files submitted via the API channel must follow the naming convention and the format described in chapter 2.

1.1.2. eDesk

Method of transmitting reports via eDesk is available at the following link: <https://edesk.apps.cssf.lu/dpaaif/edesk-dpaaif/home>.

The eDesk module allows:

- the filling of the questionnaire;
- the submission of the questionnaire;
- the consultation of submitted questionnaire;
- the PDF export of data contained in the questionnaire;

The questionnaire must be submitted by a user with the "Authorized Director" specific role, linked to the concerned entity. All information concerning specific role within an entity can be found in our dedicated user guide, accessible on the eDesk homepage.

1.2. Technical format for API channel

1.2.1. Input files

The data concerning the Self-assessment questionnaire PDAOFI shall be filed in the JSON (JavaScript Object Notation) format.

The schema concerning the input file is available at:

Publication date	Version	Structure	
10/01/2025	1.0.0	schema	Initial version

The input file must be transmitted via a compressed .zip format containing one single .json file. The annex documents of the questionnaire cannot be added to the zip archive and must be manually uploaded through the eDesk form.

The zip archive must comply with the following naming convention:

TYPFIL-EIIIIIII-YYYY-TYP-UUID.ext

where

Code	Meaning	Structure	Authorised values
TYPFIL	File type	Char(6)	'DPAAIF' for SAQ PDAOFI
-	Separator	Char(1)	Constant '-'
E	Entity type of the PDAOFI	Char(1)	'P'
IIIIIII	Identification number of the PDAOFI	Number(8)	0000001...9999999
-	Separator	Char(1)	Constant '-'
YYYY	Reference year	Number(4)	Reporting period
-	Separator	Char(1)	Constant '-'
UUID	Unique identifier (ReportUid)	UUID format	Unique identifier following the rfc 4122 norm
.ext	Extension	Char(5)	' .zip'

Example:

DPAAIF-P99999999-2024-41c13c08-ef59-4183-870f80d7a00c54e8.zip

UUID specifications The UUID is necessary to uniquely identified a report concerning PDAOFI. It is also used to prevent a file to be process several times and to allow the CSSF to share a processing order of files concerning a same PDAOFI. A ZIP file with the same UUID of another ZIP file concerning the same PDAOFI will be rejected with the error DPAAIF007 (cf. chapter 1.2.2). The expected

format of an UUID is: XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX where each X data can have a value from 0 to 9 or from a to f (hexadecimal authorised characters). You can find more information about the UUID on: [Universally Unique Identifier \(UUID\)](#)

No specific naming convention will apply for the JSON file included in the zip archive, but it is recommended to use report.json.

1.2.2. CSSF feedback files

The data contained in the report files are subject to a list of controls.

Code	Message	Severity
DPAAIF000	The report has been received and is available from eDesk for submission at the CSSF	INFO
DPAAIF001	The archive is corrupted	ERROR
DPAAIF002	The archive name is invalid. The expected naming convention must follow the regex: DPAAIF-(?<entityCssfCode>P[0-9]{8})-(?<referencePeriod>[1-9][0-9]{3})-(?<reportUid>[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}).zip	ERROR
DPAAIF003	The archive must contain a single JSON file	ERROR
DPAAIF004	The archive size cannot exceed 20 MB	ERROR
DPAAIF005	The submitter is not authorized to transmit a report for this entity	ERROR
DPAAIF006	The entity defined in the archive name is not in scope for the reference period	ERROR
DPAAIF061	The report for the entity and the reference period defined in the archive name is already submitted via eDesk	ERROR
DPAAIF007	The archive with reportUid %s has already been received for this entity	ERROR
DPAAIF008	JSON file does not respect JSON Schema	ERROR
DPAAIF009	The unique identifier '%s' in the JSON file is different from the unique identifier used in the archive name	ERROR
DPAAIF010	Due to the condition '%s', the field '%s' value must not be provided	ERROR

These controls are performed automatically when a report file is submitted.

If at least one error with the "ERROR" severity is raised, the file is rejected and there is no pre-filling of the questionnaire.

A feedback file is automatically transmitted when an input file is validated.

This feedback file complies with the following naming convention:

SourceFileName_FEEDBACK_TrackingCode.ext

where

Code	Meaning	Structure	Authorised values
SourceFileName	Name of the input file	Char(N)	Submitted file name – Refer to the ZIP File name structure in section 1.2.1
_	Separator	Char(1)	Constant `_'
FEEDBACK	File type	Char(8)	FEEDBACK (constant)
TrackingCode	Unique identifier	Char(19)	DPAAIFYYYYMMDDXXXXXX
.ext	Extension	Char(5)	`.json`

Example:

DPAAIF-P99999999-2024-41c13c08-ef59-4183-870f80d7a00c54e8.zip_FEEDBACK_DPAAIF2025010100001.json

The schema concerning the feedback file is available at:

Publication date	Version	Structure	
10/01/2025	1.0.0	schema	Initial version

A feedback file contains three information:

- A status at the report level (ACCEPTED or REJECTED);
- A list of the ReportUids of the recently processed file concerning the PDAOFI;
- A list of the error(s) raised in case of rejection of the file

2. Contact information

If you have any questions, please contact edesk@cssf.lu