

Description of ARM services - test environment

1. File names should meet the following requirements:

1.1. The file name should comply with the convention defined by the Financial Supervision Commission

(<senderLEI(20)>_<DDMMYYYY>_<Sequence(4)>.xml,

Where

senderLEI(20) is the LEI code of the ARM;

DDMMYYYY is the date of submitting the file;

Sequence(4) is the reference number of the submitted file for the day.

For example: XXXXXXXXXXXXXXXXXXXXXXXX_19032018_0001.xml). A file with the same name should not have been previously submitted.

1.2. The files should be signed with the same signature as the one, used in the registration.

2. The System provides three different RESTful services for uploading and downloading of files. They are shown below. The object used with the methods are shown as C# declarations. Convertors such as <http://csharp2json.io/> can be used for C#-to-JSON conversion.

FINANCIAL SUPERVISION COMMISSION

2.1. File Upload. The service **UploadTransactionFile** provides the ability to upload a signed file into the system.

UploadTransactionFile service

Endpoint: <https://mifir.fsc.bg:11443/api/TransactionFiles/Import>

Request: POST with JSON object based on the class ImportSingleXMLRequest:

```
public class ImportSingleXMLRequest
{
    public StandartFile File { get; set; }

    public string ReportingUnitCode { get; set; }
}

public class StandartFile
{
    public byte[] Document { get; set; }

    public string FileName { get; set; }
}
```

Request Example:

```
{
  "file": {
    "document": "AAAAAAAAAAAAAAAA==",
    "fileName": "abcd.xml"
  },
  "reportingUnitCode": "ACCDDSSS"
}
```

Response: JSON object based on the class SingleFileImportResponse:

```
public class SingleFileImportResponse: StatusResponse //at the end of the document
{
    public string Num { get; set; }
}

public class StatusResponse
{
    public StatusType StatusType { get; set; }
    public List<string> Message { get; set; } // in case of StatusType.Error,
contents list of errors
}

public enum StatusType
{
    Error,
    Success
}
```

FINANCIAL SUPERVISION COMMISSION

2.2. Providing a list of downloadable files by date (from / to). The service **GetFeedbackFilesByDate** will return a list of file names with validation feedback. The list will be adjustable according to the set period (including from date to date). If the service is requested only by “From date”, it will return a list with all the files, which have been generated from the set date on. Similarly, if the service is requested only by “to date”, it should return a list of all the files, generated up to the set date, including the very set date.

GetFeedbackFilesByDate service

Endpoint: <https://mifir.fsc.bg:11443/api/TransactionFiles/GetFeedbackFilesByDate>

Request: POST with JSON object based on the class GetFeedbackFilesByDateAndLeiRequest:

```
public class GetFeedbackFilesByDateAndLeiRequest
{
    public DateTime? From { get; set; }
    public DateTime? To { get; set; }
    public string LEI { get; set; }
}
```

Request Example:

```
{
  "from": "2018-10-24T00:00:00+00:00",
  "to": "2018-10-24T00:00:00+00:00",
  "lei": "1234567890987654321"
}
```

Response: JSON object based on the class GetFeedbackFilesByDateResponse:

```
public class GetFeedbackFilesByDateResponse: StatusResponse //at the end of the
document
{
    public List<string> FeedbackFilesNames { get; set; }
}

public class StatusResponse
{
    public StatusType StatusType { get; set; }
    public List<string> Message { get; set; } // in case of StatusType.Error,
contents list of errors
}

public enum StatusType
{
    Error,
    Success
}
```

FINANCIAL SUPERVISION COMMISSION

2.3. Downloading a file with a particular name. The service **DownloadFeedbackFile** allows the downloading of a file with a particular name.

DownloadFeedbackFile service

Endpoint: <https://mifir.fsc.bg:11443/api/TransactionFiles/DownloadFeedbackFile>

Request: POST with JSON object based on the class DownloadOpenFileByLEIRequest:

```
public class DownloadOpenFileByLEIRequest
{
    public string FileName { get; set; }
    public string LEI { get; set; }
}
```

Request Example:

```
{
  "fileName": "aaaa.xml",
  "lei": "1234567890987654321"
}
```

Response: JSON object based on the class DownloadOpenFileResponse:

```
public class DownloadOpenFileResponse: StatusResponse //at the end of the document
{
    public byte[] FileBytes { get; set; }
}

public class StatusResponse
{
    public StatusType StatusType { get; set; }
    public List<string> Message { get; set; } // in case of StatusType.Error,
    contents list of errors
}

public enum StatusType
{
    Error,
    Success
}
```

FINANCIAL SUPERVISION COMMISSION

3. Example for work scenarios.

- ARM sends a signed file through the **UploadTransactionFile** service.
- TREM Portal validates the file. The file goes through a successful XSD validation (success). The service returns a response with an incoming reference number of the document.
- At a certain hour the ARM addresses the TREM Portal with a request for newly generated files with the current day date.
- The System returns a list of one generated file - a response containing the file validation status, which has been received the previous day.
- ARM requests downloading the file.
- On the next day:**
- ARM sends a signed file through the **UploadTransactionFile** service.
- TREM Portal validates the file. The file goes through an unsuccessful XSD validation (error).
- At a certain hour the ARM addresses the TREM Portal with a request for newly generated files with the current day date.
- The System returns a list of two files: a generated file – a response containing the file validation status, which has been received the previous day, as well as a file containing a definition of the XSD validation errors, delivered earlier on the same day.
- ARM sends two consecutive requests for downloading the two files (by name of the file).