



ธนาคารแห่งประเทศไทย
BANK OF THAILAND

Data Acquisition API Specification

ระบบบริการรับส่งข้อมูลอิเล็กทรอนิกส์ของธนาคารแห่งประเทศไทยในรูปแบบ API

มีนาคม 2565

Document Information

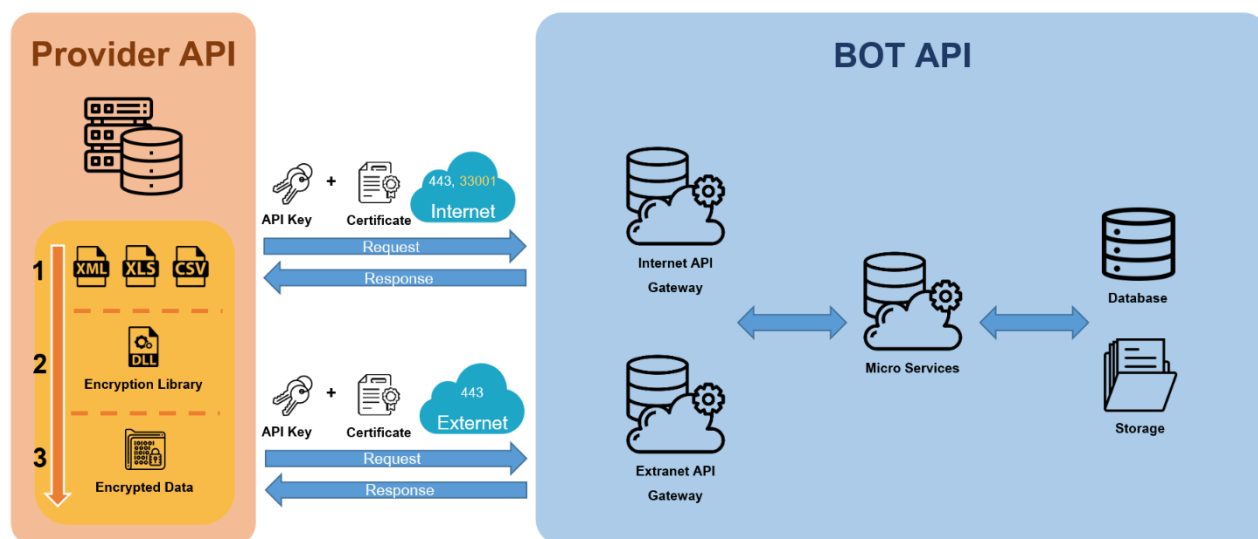
Revision History

Version number	Released Date	Effective Date	Summary of changes	Revision marks
1.0	16 กรกฎาคม 2564			
1.1	29 ตุลาคม 2564		- System Requirement - Sequence Diagram - Required Field	
1.2	14 มีนาคม 2565		- URL Internet ชุด Production - Network Port - Download BOT Public Key - API – Cross Validation Group	Changes from 1.1 in red.

Table of Contents

I. API Specification	4
รายการ API ที่ให้บริการ.....	5
II. Authentication (การพิสูจน์ตัวตน)	6
III. System Requirement	7
1. Operating System.....	7
2. Hardware	7
3. Software	7
4. Network.....	7
5. Bandwidth	8
IV. Sequence Diagram	9
1. Normal Submission	9
2. FASP Submission.....	10
3. View Submission Logs	11
V. API – Information	12
1. API – Submission Requirement.....	12
2. API – Submission Validate.....	15
3. API – Submission	17
4. API – Submission FASP Ticket	19
5. API – Submission FASP	24
6. API – Submission Logs.....	27
7. API – Submission Error Log	33
8. API – Cross Validation Groups	34
9. API – Cross Validation Logs	36
10. API – Cross Validation Error Log.....	40
VI. Appendix	41
1. Appendix A: การเตรียมไฟล์ก่อนการส่งข้อมูล	41
2. Appendix B: การส่งไฟล์ด้วยวิธี HTTP Protocol.....	44
3. Appendix C: การส่งไฟล์ด้วยวิธี File Transfer Protocol	44

I. API Specification



รูปแบบ API ของธนาคารแห่งประเทศไทย

ขั้นตอนการทำงานในการเชื่อมต่อระหว่างองค์กรที่ส่งข้อมูลกับ ธนาคารแห่งประเทศไทย (ธปท.) ผ่านระบบ Data Acquisition API จะเชื่อมต่อด้วย Certificate ที่รับรองโดย Trusted Third-party Certificate Authority (CA) ที่ผ่านมาตรฐานด้านความปลอดภัย (ISO27001) ซึ่งปัจจุบันรองรับ Certificate ที่ออกโดย บริษัท ไทยดิจิทัล ไอดี จำกัด (Thai Digital ID Co.,Ltd. – TDID) และ INET (Internet Thailand) ทั้งนี้ Certificate Authority อื่น ขอให้ติดต่อ ธปท. ล่วงหน้า

โดยการส่งข้อมูลนั้น ผู้ส่งจะต้องทำการเข้ารหัสไฟล์ข้อมูลด้วย Library ที่ทาง ธปท. พัฒนาขึ้น ซึ่งเป็นการเข้ารหัสข้อมูลแบบ RSA และ AES256 โดยสามารถดาวน์โหลด Library ผ่านทางหน้า BOT Website (URL : <https://www.bot.or.th/Thai/Statistics/DataManagementSystem/Standard/DMSDA/Pages/DMSDA.aspx>)

รายการ API ที่ให้บริการ

Environment	Channel	URL
Production	Internet	https://apigw2.bot.or.th/bot/
	Extranet	https://apixgw2.bot.or.th/bot/
IWT	Internet	https://apigw2.bot.or.th/bot-iwt/
	Extranet	https://apixgw2.bot.or.th/bot-iwt/

API	Endpoint	Internet	Extranet
API – Submission Requirement	/partner/da-submission-requirements	/	/
API – Submission Validate	/partner/da-submission-validate	/	/
API – Submission	/partner/da-submission	/	/
API – Submission FASP Ticket	/partner/da-submission-fasp-ticket	/	
API – Submission FASP	/partner/da-submission-fasp	/	
API – Submission Logs	/partner/da-submission-logs	/	/
API – Submission Error Log	/partner/da-submission-error-log	/	/
API – Cross Validation Groups	/partner/da-cross-validation-groups	/	/
API – Cross Validation Logs	/partner/da-cross-validation-logs	/	/
API – Cross Validation Error Log	/partner/da-cross-validation-error-log	/	/

II. Authentication (การพิสูจน์ตัวตน)

ในการเรียกใช้ API องค์กรต้องดำเนินการลงทะเบียนไว้กับระบบ BOT API

- Production : <https://apiportal.bot.or.th/bot/partner/>
- IWT : <https://apiportal.bot.or.th/bot-iwt/partner/>

1. **Certificate:** ใบรับรองอิเล็กทรอนิกส์ เพื่อยืนยันตัวตนว่าเป็นองค์กรนั้นจริงและได้ลงทะเบียนกับระบบ BOT API
2. **HTTP Header:** HTTP Header ที่ได้รับจากการลงทะเบียน BOT API เพื่อใช้ในการ Request API
 - X-IBM-Client-Id
 - X-IBM-Client-Secret

```
HttpClientHandler handler = new HttpClientHandler();
handler.ClientCertificates.Add(new X509Certificate2(@"\path\to\certificate.pfx", "password"));
handler.SslProtocols = SslProtocols.Tls12;

HttpClient client = new HttpClient(handler);
client.DefaultRequestHeaders.Add("X-IBM-Client-Id", "client-id");
client.DefaultRequestHeaders.Add("X-IBM-Client-Secret", "client-secret");
```

III. System Requirement

1. Operating System

- Windows Server 2012 – 64 bits
- Linux Version ที่ Support .NET Core 3.1 เป็นต้นไป

2. Hardware

- RAM ไม่ต่ำกว่า 16 GB
- CPU ไม่ต่ำกว่า Dual Core 1.8 GHz (ทั้งนี้ขึ้นอยู่กับ ปริมาณงานบน Server ซึ่งองค์กรพิจารณาให้เหมาะสม)

3. Software

- Library “DAEncryptionLib.dll” ที่รพท. เผยแพร่บน BOT Website
<https://www.bot.or.th/Thai/Statistics/DataManagementSystem/Standard/DMSDA/Pages/DMSDA.aspx> (สำหรับการเข้ารหัสไฟล์ก่อนส่งข้อมูล)

4. Network

- Internet
 - Port 443
 - ใช้ URL: apigw2.bot.or.th ซึ่งสามารถ Resolve DNS Name ได้จาก Public DNS Domain
 - หากเครื่อง Server ติดตั้งอยู่ต่างประเทศต้องแจ้ง IP เพื่อให้ทาง รพท. ดำเนินการทำ White List
- Extranet
 - Port 443
 - ใช้ URL: apigw2.bot.or.th ซึ่งที่ Client Machine จำเป็นต้อง fix host ของ apigw2.bot.or.th ไปยัง IP Address 192.168.197.142

5. Bandwidth

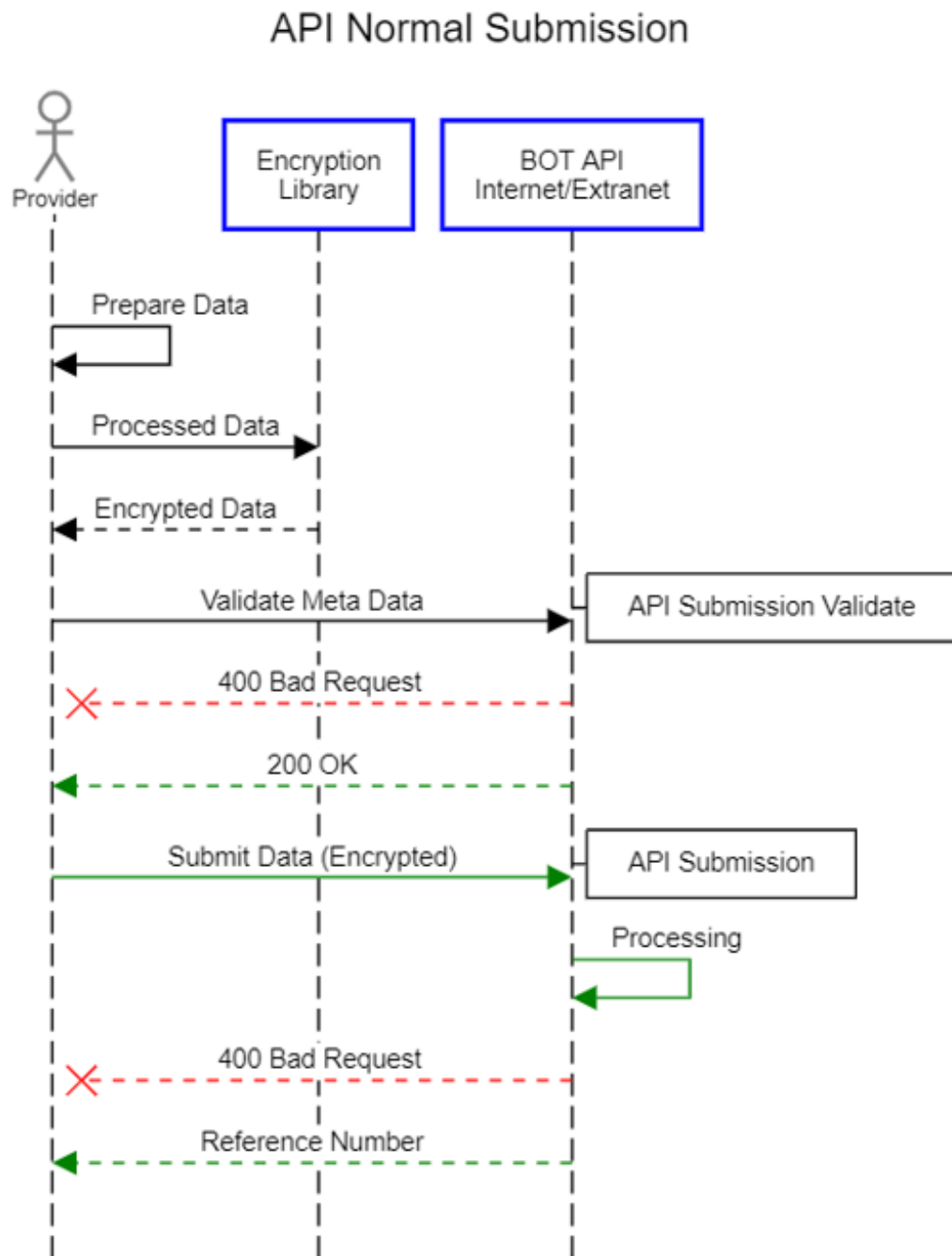
- สถาบันการเงินควรต้องประเมินขนาด Bandwidth ของเครือข่ายจากผู้ให้บริการให้เพียงพอกับปริมาณข้อมูลที่ส่งไปยัง ธปท. และครอบคลุมกับระบบงานอื่นๆ ของสถาบันการเงิน
- ตัวอย่างประมาณ Bandwidth และเวลาในการส่งข้อมูล 1 ไฟล์ โดยตารางเป็นการส่งข้อมูลภายใต้เงื่อนไขดังนี้
 - File Size 1 GB

หมายเหตุ : กรณีการส่งข้อมูลผ่านช่องทาง Internet เวลาอาจเพิ่มขึ้นตามความหนาแน่นของ Traffic ในขณะนั้น

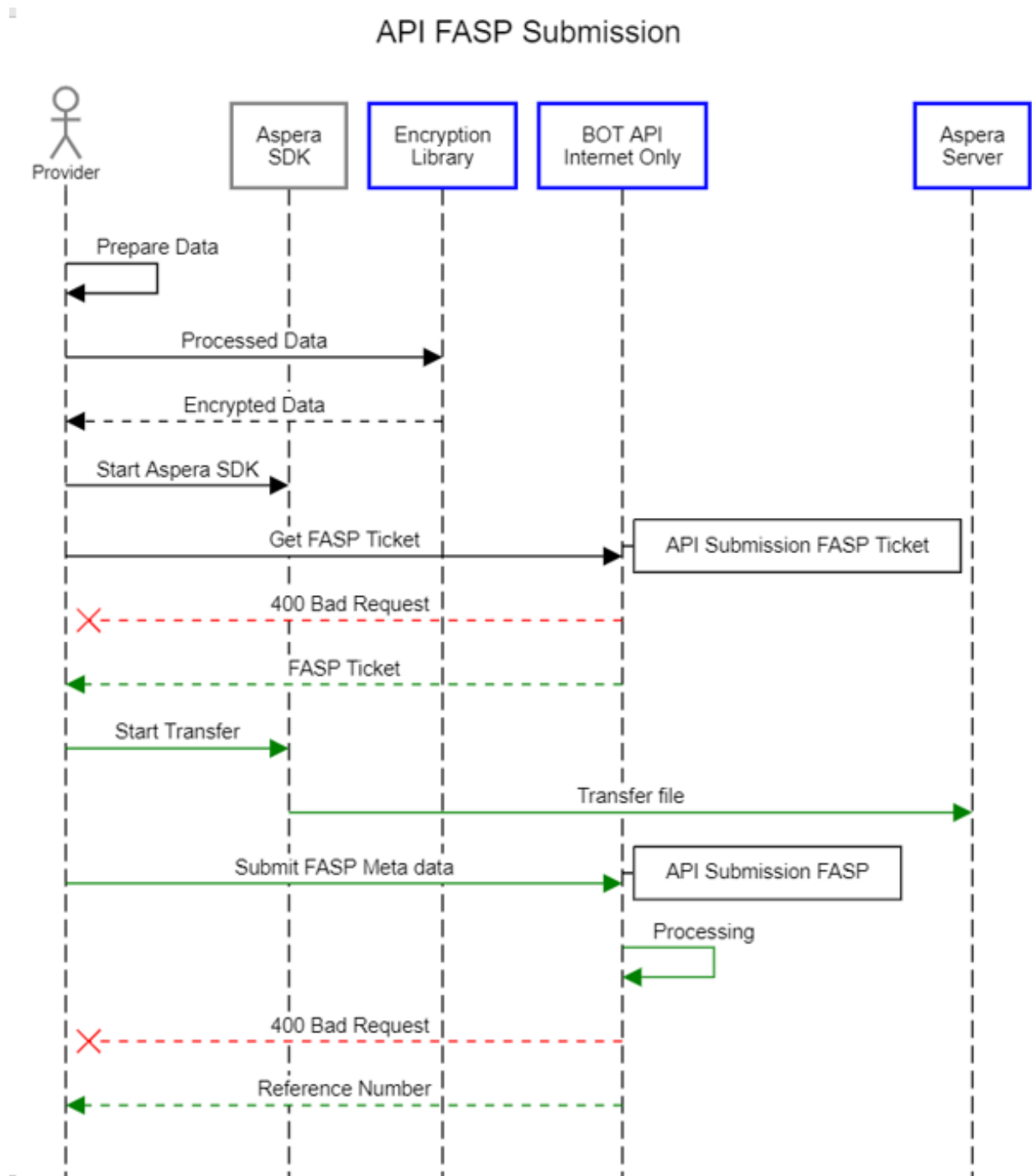
Bandwidth	Estimated File Transfer Time (Minutes)	
	FASP	Normal (TCP)
1 Mbps	144	144.8
2 Mbps	71.7	72.5
10 Mbps	14.3	15
15 Mbps	9.6	10.2
20 Mbps	7.2	7.8
100 Mbps	N/A	1.6

IV. Sequence Diagram

1. Normal Submission

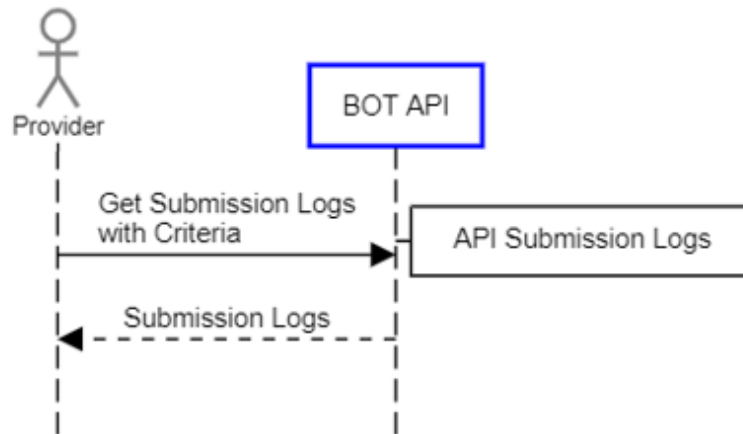


2. FASP Submission



3. View Submission Logs

API Submission Logs



V. API – Information

รายละเอียดการใช้งาน BOT API เพื่อใช้ในการส่งข้อมูลมายังระบบ Data Acquisition

Require Field Description

- M: Mandatory field.
- C: Mandatory field with condition.
- O: Optional.

1. API – Submission Requirement

API สำหรับการเรียกดูรายละเอียดในการส่งของแต่ละชุดข้อมูล ก่อนที่จะส่งข้อมูลมายัง ระบบ. ด้วยรหัสชุดข้อมูล โดยไม่จำเป็นต้องเรียกทุกครั้งก่อนส่ง (เรียกกรณีที่ต้องการค้นหารหัสชุดข้อมูล)

- HTTP Method:

HTTP Method	GET
-------------	-----

- Response HTTP Status Code:

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- Response Body

Name	Data Type	Data Length	Description	Example
Dataset.Id	String	20	รหัสชุดข้อมูล	DS_000006
Dataset.Name	String	500	ชื่อชุดข้อมูล	Data Entity Account Detail
Dataset.Description	String	500	รายละเอียดชุดข้อมูล	ข้อมูลรายละเอียดสัญญา
Dataset.Abbreviation	String	20	ชื่อย่อชุดข้อมูล	DER_CACD
Dataset.SubjectArea.Id	Integer		รหัสหัวข้อข้อมูล	99999

Dataset.SubjectArea.Name	String	500	ชื่อหัวข้อข้อมูล	RDT Credit
Dataset.SubjectArea.Description	String	500	รายละเอียดหัวข้อข้อมูล	ข้อมูลสินเชื่อ
Dataset.EntityGroup.Id	String	20	รหัส Entity Group	EN_000001
Dataset.EntityGroup.Name	String	500	ชื่อ Entity Group	Account
Dataset.EntityGroup.Description	String	500	รายละเอียด Entity Group	Account
Dataset.Granularity.Id	String	20	รหัสความถี่ของเนื้อข้อมูล	MT
Dataset.Granularity.Name	String	500	ชื่อความถี่ของเนื้อข้อมูล	Monthly
Dataset.Frequency.Id	String	20	รหัสความถี่การส่งข้อมูล	MT
Dataset.Frequency.Name	String	500	ชื่อความถี่การส่งข้อมูล	Monthly
Dataset.Security.Id	Integer		รหัสความปลอดภัยข้อมูล	9
Dataset.Security.Name	String	500	ชื่อความปลอดภัยข้อมูล	Certificate
Provider.Id	String	40	รหัสองค์กร	999
Provider.Name	String	500	ชื่อองค์กร	Example Bank
Provider.NameTH	String	500	ชื่อองค์กรภาษาไทย	ธนาคารทดสอบ จำกัด
EffectiveDate	DateTime		วันที่เริ่มต้น	2022-01-01T00:00:00
EndDate	DateTime		วันที่สิ้นสุด	null

Response Body JSON Example

```
[
  {
    "Dataset": {
      "Id": "DS_000006",
      "Name": "Data Entity Account Detail",
      "Description": "ข้อมูลรายละเอียดสัญญา",
      "Abbreviation": "DER_CACD",
      "SubjectArea": {
        "Id": 999,
        "Name": "RDT Credit",
        "Description": "ข้อมูลสินเชื่อ"
      },
      "Granularity": {
        "Id": "MT",
        "Name": "Monthly"
      },
      "Frequency": {
        "Id": "MT",
        "Name": "Monthly"
      }
    }
  }
]
```

```

    },
    "Security": {
        "Id": 9,
        "Name": "Certificate"
    }
},
"Provider": {
    "Id": "999",
    "Name": "Example Bank",
    "NameTH": "ธนาคารทดสอบ จำกัด"
},
"EffectiveDate": "2022-01-01T00:00:00",
"EndDate": null,
}]

```

- **Sample Code:**

```

using var response = await client.GetAsync("https://[redacted]/partner/da-submission-requirements");
Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());

```

2. API – Submission Validate

API สำหรับใช้ในการตรวจ Meta Data ก่อนที่จะส่งข้อมูลผ่าน [API – Submission](#)

- HTTP Method:

HTTP Method	POST
-------------	------

- Request Body:

Name	Data Type	Data Length	Require	Description	Example
datasetId	String	40	M	รหัสชุดข้อมูล	DS_000006
measurementPeriod	DateTime		M	วันที่ของงวดข้อมูล	2021-06-30T00:00:00
submissionType	String	1	M	ประเภทการส่งข้อมูล <ul style="list-style-type: none">● A – Adjustment● N – Normal	A
adjustReasonId	Integer		C	รหัสเหตุผลของการ Adjustment <ul style="list-style-type: none">● 1 – Incorrect data reporting● 2 – System error● 3 – Customer request● 4 – Auditor request● 5 – BOT request	null
adjustSubject	String	500	C	หัวข้อของการ Adjustment	แก้ไข Country to Invest
adjustContent	String	4000	C	เนื้อหาของการ Adjustment	แก้ไข Country to Invest สัญญาเลขที่ 12345678 จาก GR เป็น GB

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Sample Code:**

```
var obj = new
{
    datasetId = "DS_000460",
    measurementPeriod = new DateTime(2021, 6, 30),
    submissionType = "N",
    // Uncomment below if submission type = 'A'
    // adjustReasonId = 1,
    // adjustSubject = "Test Adjustment Subject",
    // adjustContent = "Test Adjustment Content",
};

var sc = new StringContent(JsonConvert.SerializeObject(obj), Encoding.UTF8, "application/json");

using var response = await client.PostAsync("https://[REDACTED]/partner/da-submission-validate", sc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());
```


3. API – Submission

API สำหรับใช้ในการส่งข้อมูลด้วย HTTP Protocol (multipart/form-data)

- HTTP Method:

HTTP Method	POST
-------------	------

- Request Form Data:

Name	Data Type	Data Length	Require	Description	Example
file	File		M	ไฟล์ข้อมูลที่ต้องการส่ง	
datasetId	String	40	M	รหัสชุดข้อมูล	DS_000006
measurementPeriod	DateTime		M	วันที่ของงวดข้อมูล	2021-06-30T00:00:00
submissionType	String	1	M	ประเภทการส่งข้อมูล <ul style="list-style-type: none">● A – Adjustment● N – Normal	A
adjustReasonId	Integer		C	รหัสเหตุผลของการ Adjustment <ul style="list-style-type: none">● 1 – Incorrect data reporting● 2 – System error● 3 – Customer request● 4 – Auditor request● 5 – BOT request	1
adjustSubject	String	500	C	หัวข้อของการ Adjust	แก้ไข Country to Invest
adjustContent	String	4000	C	เนื้อหาของการ Adjust	แก้ไข Country to Invest สัญญาเลขที่ 12345678 จาก GR เป็น GB

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Response Data:**

Name	Data Type	Data Length	Description	Example
Reference Number	long		เลข Reference ในการส่งข้อมูล	12345

Body JSON Example

12345

Sample Code:

```
using var mc = new MultipartFormDataContent();

string filePath = @"C:\_Temp\MRDTCredit999_20220131_ACD.csv.bot";

using var sc = new StreamContent(File.OpenRead(filePath));

mc.Add(sc, "file", Path.GetFileName(filePath));
mc.Add(new StringContent("DS_000006", Encoding.UTF8), "datasetId");
mc.Add(new StringContent("2021-01-31", Encoding.UTF8), "measurementPeriod");
mc.Add(new StringContent("N", Encoding.UTF8), "submissionType");
// Uncomment below if submission type = 'A'
// mc.Add(new StringContent("แก้ไขสกุลเงิน", Encoding.UTF8), "adjustSubject");
// mc.Add(new StringContent("แก้ไขสกุลเงินสัญญาเลขที่ 12345678 จาก EUR เป็น GBP", Encoding.UTF8), "adjustContent");
// mc.Add(new StringContent(1.ToString(), Encoding.UTF8), "adjustReasonId");

using var response = await client.PostAsync("https://[redacted]/partner/da-submission", mc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());
```

4. API – Submission FASP Ticket

API สำหรับใช้ในการขอ Ticket เพื่อส่งข้อมูลผ่านทาง Aspera (Protocol FASP- File Transfer Aspera) (**เฉพาะ ช่องทาง Internet เท่านั้น)

- HTTP Method:

HTTP Method	POST
-------------	------

- Request Body:

Name	Data Type	Data Length	Require	Description	Example
datasetId	String	40	M	รหัสชุดข้อมูล	DS_000006
measurementPeriod	DateTime		M	วันทิ้งวัดข้อมูล	2022-01-31T00:00:00
submissionType	DateTime		M	ประเภทการส่งข้อมูล <ul style="list-style-type: none"> • A – Adjustment • N – Normal 	A
adjustReasonId	Integer		C	รหัสเหตุผลของการ Adjustment <ul style="list-style-type: none"> • 1 – Incorrect data reporting • 2 – System error • 3 – Customer request • 4 – Auditor request • 5 – BOT request 	1
adjustSubject	String	500	C	หัวข้อของการ Adjustment	แก้ไข Country to Invest
adjustContent	String	4000	C	เนื้อหาของ การ Adjustment	แก้ไข Country to Invest สัญญาเลขที่ 12345678 จาก GR เป็น GB
fileName	String	200	O	ชื่อไฟล์	null
fileLocation	String	500	O	ที่อยู่ของไฟล์บน Aspera	null

Request Body JSON Example

```
{
  "datasetId": "DS_000006",
  "measurementPeriod": "2022-01-31T00:00:00",
  "submissionType": "A",
  "adjustReasonId": 1,
  "adjustSubject": "แก้ไข Country to Invest "
  "adjustContent": "แก้ไข Country to Invest สัญญาเลขที่ 12345678 จาก GR เป็น GB",
  "fileName": null,
  "fileLocation": null
}
```

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
204	No Content
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Response Body:**

Name	Data Type	Data Length	Description	Example
transfer_specs	Object Array		An array of transfer specifications that are allowed by the specified tokens.	
transfer_spec	Object		The transfer specification.	
paths	Object Array		Array of JSON objects containing source and, if specified, destination paths.	
source	String		The path for the source file or directory, relative to the source root.	

destination	String		Destination path for the file or directory that is being transferred, relative to the destination root. If no destination is specified, the content is transferred to the destination root.	
source_root	String		The source root directory of the files.	
destination_root	String	500	The destination root directory.	/DataFileAPI/249549
token	String		The token string.	ATM3_AEAO2fsqmQJjW3E KWaNw7EcSaE_9RtWYYIP kL_leHuPmrAAHzdOdwBR 6iAK0Dh2116lt0_3MTA
direction	String		For uploads, the value is send.	
cipher	String		The weakest cipher that is allowed for the transfer.	aes-256
Tags	Object		Tags.	null
rate_policy_allowed	String		The most aggressive rate policy that is allowed for the transfer.	fixed
rate_policy	String		The congestion control behavior.	fair
target_rate_kbps	Integer		The desired speed of the transfer, in Kbps.	100000
min_rate_kbps	Integer		The minimum transfer rate, in Kbps.	0
remote_host	String		IP or fully qualified domain name of the remote server.	apift-iwt.bot.or.th
remote_user	String		The system user that authenticates the SSH connection for the transfer, derived from the request authentication method.	johndoe
sshfp	String		The node's SSH fingerprint, if it is configured in the node's aspera.conf.	null
ssh_port	Integer		The server's TCP port that is listening for SSH connections. FASP® initiates transfers through SSH.	33001
fasp_port	Integer		The UDP port for FASP to use. The default value works for most situations; however, it can be changed to satisfy firewall requirements.	33001

http_fallback	Boolean		If set to true, the transfer attempts an HTTP or HTTPS transfer if FASP transfer errors.	false
https_fallback_port	Integer		The port that is used for HTTPS transfers.	0

Response Body JSON Example

```
{
  "transfer_specs": [
    {
      "transfer_spec": {
        "paths": [],
        "source_root": "",
        "destination_root": "/DataFileAPI/249549",
        "token":
"ATM3_AEAO2fsqmQJjW3EKWaNw7EcSaE_9RtWYYIPkL_leHuPmrAAHzdOdwBR6iAK0Dh2116lt0_3MTA",
        "direction": "send",
        "cipher": "aes-256",
        "tags": null,
        "rate_policy_allowed": "fixed",
        "rate_policy": "fair",
        "target_rate_kbps": 100000,
        "min_rate_kbps": 0,
        "remote_host": "apift-iwt.bot.or.th",
        "remote_user": "johndoe",
        "sshfp": null,
        "ssh_port": 33001,
        "fasp_port": 33001,
        "http_fallback": false,
        "https_fallback_port": 0
      }
    }
  ]
}
```

- **Sample Code:**

```
var obj = new
{
    datasetId = "DS_000006",
    measurementPeriod = new DateTime(2021, 6, 30),
    submissionType = "N",
    // Uncomment below if submission type = 'A'
    // adjustReasonId = 1,
    // adjustSubject = "Test Adjustment Subject",
    // adjustContent = "Test Adjustment Content",
};

var sc = new StringContent(JsonConvert.SerializeObject(obj), Encoding.UTF8, "application/json");

using var response = await client.PostAsync("https://[redacted]/partner/da-submission-fasp-ticket", sc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());
```

5. API – Submission FASP

API สำหรับใช้ในการส่ง Meta Data หลังจากส่งข้อมูลผ่าน Aspera (Protocol FASP- File Transfer Aspera) สำเร็จ (** เฉพาะช่องทาง Internet เท่านั้น)

- HTTP Method:

HTTP Method	POST
-------------	------

- Request Body:

Name	Data Type	Data Length	Req.	Description	Example
datasetId	String	40	M	รหัสชุดข้อมูล	DS_000006
measurementPeriod	DateTime		M	วันที่วัดข้อมูล	2021-06-30T00:00:00
submissionType	DateTime	1	M	ประเภทการส่งข้อมูล <ul style="list-style-type: none"> ● A – Adjustment ● N – Normal 	N
adjustReasonId	Integer		C	รหัสเหตุผลของการ Adjustment <ul style="list-style-type: none"> ● 1 – Incorrect data reporting ● 2 – System error ● 3 – Customer request ● 4 – Auditor request ● 5 – BOT request 	null
adjustSubject	String	500	C	หัวข้อของการ Adjustment	null
adjustContent	String	8000	C	เนื้อหาของการ Adjustment	null
fileName	String	200	M	ชื่อไฟล์	ไฟล์ทดสอบ.xml.bot
fileLocation	String	500	M	ที่อยู่ของไฟล์บน Aspera	/dev/fs-api/999999

Request Body JSON Example

```
{
  "datasetId": "DS_000006",
  "measurementPeriod": "2021-06-30T00:00:00",
  "submissionType": "N",
  "adjustReasonId": null,
  "adjustSubject": null,
  "adjustContent": null,
  "fileName": "ไฟล์ทดสอบ.xml.bot ",
  "fileLocation": "/dev/fs-api/9999999"
}
```

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Response Data:**

Name	Data Type	Data Length	Description	Example
Reference Number	long		เลข Reference การส่งข้อมูล	12345

Response Body JSON Example

```
12345
```

- **Sample Code:**

```
var obj = new
{
    datasetId = "DS_000006",
    measurementPeriod = new DateTime(2021, 6, 30),
    submissionType = "N",
    // Uncomment below if submission type = 'A'
    // adjustReasonId = 1,
    // adjustSubject = "Test Adjustment Subject",
    // adjustContent = "Test Adjustment Content",
    fileName = "035DS_LAR20210331.xml.bot",
    fileLocation = "/DataFileAPI/999999"
};

var sc = new StringContent(JsonConvert.SerializeObject(obj), Encoding.UTF8, "application/json");

using var response = await client.PostAsync("https://[redacted]/partner/da-submission-fasp", sc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());
```

6. API – Submission Logs

API สำหรับใช้ในการตรวจสอบสถานะการส่งข้อมูล และสถานะการประมวลผลข้อมูล

- **HTTP Method:**

HTTP Method	POST
-------------	------

- **Request Body:**

Name	Data Type	Data Length	Require	Description	Example
datasetIdList	String Array	40	O	Array รหัสชุดข้อมูล	DS_000006, DS_000086
measurementPeriodFrom	DateTime		M	วันที่ของงวดข้อมูลเริ่มต้น	2022-01-31T00:00:00
measurementPeriodTo	DateTime		O	วันที่ของงวดข้อมูลสิ้นสุด	null
submitFrom	DateTime		O	วันที่ส่งข้อมูลเริ่มต้น	2022-01-30T08:56:05.779Z
submitTo	DateTime		O	วันที่ส่งข้อมูลสิ้นสุด	2022-04-30T08:56:05.779Z
submissionType	String Array	1	O	Array ประเภทการส่งข้อมูล <ul style="list-style-type: none"> ● A – Adjustment ● N – Normal 	A,N
status	Integer Array		O	Array รหัสสถานะการส่งข้อมูล <ul style="list-style-type: none"> ● 1 – Received ● 2 – Passed Basic Validation ● 3 – Failed Basic Validation ● 5 – Passed Complex Processing ● 6 – Failed Complex Processing ● 9 – Cancelled 	1,2,3,5,6,9

refNo	Long Array		O	Array of Reference Number	null
-------	---------------	--	---	------------------------------	------

Request Body JSON Example

```
{
  "datasetIdList": [
    "DS_000006",
    "DS_000086"
  ],
  "measurementPeriodFrom": "2022-01-31T00:00:00",
  "measurementPeriodTo": null,
  "submitFrom": "2022-01-30T08:56:05.779Z",
  "submitTo": "2022-04-30T08:56:05.779Z",
  "submitType": [
    "A",
    "N"
  ],
  "status": [
    1,
    2,
    3,
    5,
    6,
    9
  ],
  "refNo": null
}
```

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

Response Body:

Name	Data Type	Data Length	Description	Example
RefNo	Long		เลข Reference Number	11111
MeasurementPeriod	DateTime		วันที่งวดข้อมูล	2022-01-31T00:00:00
Dataset.Id	String	20	รหัสชุดข้อมูล	DS_000006
Dataset.Name	String	500	ชื่อชุดข้อมูล	Data Entity Account Detail
Dataset.Description	String	500	รายละเอียดชุดข้อมูล	ข้อมูลรายละเอียดสัญญา
Dataset.Abbreviation	String	20	ชื่อย่อชุดข้อมูล	DER_CACD
Dataset.SubjectArea.Id	Integer		รหัสหัวข้อข้อมูล	99999
Dataset.SubjectArea.Name	String	500	ชื่อหัวข้อข้อมูล	RDT Credit
Dataset.SubjectArea.Description	String	500	รายละเอียดหัวข้อข้อมูล	ข้อมูลสินเชื่อ
Dataset.EntityGroup.Id	String	20	รหัส Entity Group	EN_000001
Dataset.EntityGroup.Name	String	500	ชื่อ Entity Group	Account
Dataset.EntityGroup.Description	String	500	รายละเอียด Entity Group	Account
Dataset.Granularity.Id	String	20	รหัสความถี่ของเนื้อข้อมูล	MT
Dataset.Granularity.Name	String	500	ชื่อความถี่ของเนื้อข้อมูล	Monthly
Dataset.Frequency.Id	String	20	รหัสความถี่การส่งข้อมูล	MT
Dataset.Frequency.Name	String	500	ชื่อความถี่การส่งข้อมูล	Monthly
Dataset.Security.Id	Integer		รหัสความปลอดภัยข้อมูล	9
Dataset.Security.Name	String	500	ชื่อความปลอดภัยข้อมูล	Certificate
Provider.Id	String	40	รหัสองค์กร	999
Provider.Name	String	500	ชื่อองค์กร	Example Bank
Provider.NameTH	String	500	ชื่อองค์กรภาษาไทย	ธนาคารทดสอบ จำกัด
Status.Id	Integer		รหัสสถานะการส่งข้อมูล	5
Status.Description	String	500	รายละเอียดสถานะการส่งข้อมูล	Passed Complex Processing

Status.Timestamp	DateTime		เวลาแจ้งสถานะ ล่าสุดของการส่ง ข้อมูล	2022-02-02T22:10:17
Submission.Type	String	1	ประเภทการส่งข้อมูล	N
Submission.TypeName	String	50	ชื่อประเภทการส่ง ข้อมูล	Normal
Submission.TimeStamp	DateTime		เวลาการส่งข้อมูล	2022-02-13T14:33:43
Submission.By	String	100	ชื่อผู้ส่งข้อมูล	John Doe
Submission.Uid	String	100	Username ผู้ส่ง ข้อมูล	johndoe
Submission.Channel	String	20	ช่องทางการส่งข้อมูล (Internet หรือ Extranet)	internet
Submission.FileName	String	200	ชื่อไฟล์	MCRD999_20220131_CA CD.csv.bot
Submission.FileSize	Long		ขนาดไฟล์ หน่วย เป็น Bytes	494693
Submission.Adjustment.Subject	String	500	หัวข้อการ Adjustment	Null
Submission.Adjustment.Content	String	8000	เนื้อหารายละเอียด การ Adjustment	Null
Submission.Adjustment.Reason.I d	Integer		รหัสเหตุผลการ Adjustment	Null
Submission.Adjustment.Reason. Name	String	500	ชื่อเหตุผลการ Adjustment	Null
EmptyDataset	Boolean		ข้อมูลว่างหรือไม่	false

Response Body JSON Example

```
[
  {
    "RefNo": 11111,
    "MeasurementPeriod": "2022-01-31T00:00:00",
    "Dataset": {
      "Id": "DS_000006",
      "Name": "Data Entity Account Detail",
      "Description": "ข้อมูลรายละเอียดสัญญา",
      "Abbreviation": "DER_CACD",
      "SubjectArea": {
```

```

        "Id": 99999,
        "Name": "RDT Credit",
        "Description": "ข้อมูล สินเชื่อ",
    },
    "EntityGroup": {
        "Id": "EN_000001",
        "Name": "Account",
        "Description": "Account",
    },
    "Granularity": {
        "Id": "MT",
        "Name": "Monthly"
    },
    "Frequency": {
        "Id": "MT",
        "Name": "Monthly"
    },
    "Security": {
        "Id": 9,
        "Name": "Certificate"
    }
},
"Provider": {
    "Id": "999",
    "Name": "Example Bank",
    "NameTH": "ธนาคารทดสอบ จำกัด"
},
"Status": {
    "Id": 5,
    "Description": "Passed Complex Processing",
    "Timestamp": "2022-02-02T22:10:17"
},
"Submission": {
    "Type": "N",
    "TypeName": "Normal",
    "TimeStamp": "2022-02-13T14:33:43",
    "By": "John Doe ",
    "Uid": "johndoe",
    "Channel": "Internet",

```

```

        "FileName": "MCRD999_20220131_CACD.csv.bot",
        "FileSize": 494693,
        "Adjustment": null
    },
    "EmptyDataset": false
}
]

```

- **Sample Code:**

```

// Uncomment below to filter with AND Condition.
var obj = new
{
    datasetIdList = new List<string>() { "DS_000006", "DS_000086" },
    measurementPeriodFrom = new DateTime(2018, 1, 1),
    // measurementPeriodTo = new DateTime(2018, 1, 1),
    // submitFrom: new DateTime(2018, 1, 1),
    // submitTo: new DateTime(2018, 1, 1),
    submitType = new List<string>() { "A", "N" },
    status = new List<int>() { 1, 2, 3, 5, 6, 9 },
    // refNo = new List<long>() { }
};

var sc = new StringContent(JsonConvert.SerializeObject(obj), Encoding.UTF8, "application/json");

using var response = await client.PostAsync("https://i[REDACTED]/partner/da-submission-logs", sc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());

```


7. API – Submission Error Log

API สำหรับใช้ในการ Download Error Log ของการส่งข้อมูล

- **HTTP Method:**

HTTP Method	GET
-------------	-----

- **Request Query String:**

Name	Data Type	Data Length	Require	Description	Example
refNo	Long		M	เลข Reference Number	12345

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Sample Code:**

```
long refNo = 100009275;

using var response = await client.GetAsync($"https://[REDACTED]/partner/da-submission-error-log?refNo={refNo}");

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");

using var fs = new FileStream($"C:\\_Temp\\{response.Content.Headers.ContentDisposition.FileName}", FileMode.Create);
using var sc = await response.Content.ReadAsStreamAsync();
sc.CopyTo(fs);
```

8. API – Cross Validation Groups

API สำหรับการเรียกดูรายละเอียด กลุ่มของการตรวจสอบระหว่างชุดข้อมูล (Cross Validation) ทั้งหมดที่ทางสถาบันการเงินมีสิทธิสามารถเรียกดูได้ โดยไม่จำเป็นต้องเรียกทุกครั้งก่อนส่ง (เฉพาะกรณีที่ต้องการค้นหารหัสกลุ่มของการตรวจสอบระหว่างชุดข้อมูล)

- **HTTP Method:**

HTTP Method	GET
-------------	-----

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- **Response Body**

Name	Data Type	Data Length	Description	Example
Id	String	10	รหัสกลุ่มการตรวจสอบ	6
Name	String	100	ชื่อกลุ่มการตรวจสอบ	DMS BLS vs RDT OTDM
Description	String	255	รายละเอียดกลุ่มการตรวจสอบ	Balance sheet (Credit) with RDT Credit

Response Body JSON Example

```
[
  {
    "Id": "6",
    "Name": " DMS BLS vs RDT OTDM ",
    "Description": " Balance sheet (Credit) with RDT Credit "
  }
]
```

- **Sample Code:**

```
using var response = await client.GetAsync("https://# ██████████ ██████████/partner/da-cross-validation-groups");  
Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");  
Console.WriteLine(await response.Content.ReadAsStringAsync());
```

9. API – Cross Validation Logs

API สำหรับใช้ในการตรวจสอบสถานะการตรวจสอบ Cross Validation

- HTTP Method:

HTTP Method	POST
-------------	------

- Request Body:

Name	Data Type	Data Length	Require	Description	Example
validationGroupId	String Array	10	O	Array กลุ่มของการตรวจสอบระหว่างชุดข้อมูล	1, 6, 7
measurementPeriodFrom	DateTime		M	วันที่งวดข้อมูลเริ่มต้น	2022-01-31T00:00:00
measurementPeriodTo	DateTime		O	วันที่งวดข้อมูลสิ้นสุด	2022-03-31T00:00:00
processFrom	DateTime		O	วันที่ตรวจสอบข้อมูลเริ่มต้น	2022-01-31T00:00:00
processTo	DateTime		O	วันที่ตรวจสอบข้อมูลสิ้นสุด	2022-04-01T00:00:00
status	Integer Array		O	Array รหัสสถานะการตรวจสอบระหว่างชุดข้อมูล <ul style="list-style-type: none"> ● 23 – Waiting for Related Sets ● 25 – Passed Cross Validation ● 26 – Failed Cross Validation 	23, 25, 26
refNo	String Array		O	Array เลข Cross Reference Number	XR202201310001, XR202201310002, XR202201310003
isAllVersion	Boolean		M	True – เพื่อดูผลการตรวจสอบระหว่างชุดข้อมูลทุก Version	true

				False – เพื่อดูผลการ ตรวจสอบล่าสุดของงวด ข้อมูล	
--	--	--	--	---	--

Request Body JSON Example

```
{
  "validationGroupId": [
    "1",
    "6",
    "7"
  ],
  "measurementPeriodFrom": "2022-01-31T00:00:00",
  "measurementPeriodTo": "2022-03-31T00:00:00",
  "processFrom": "2022-01-31T00:00:00",
  "processTo": "2022-04-01T00:00:00",
  "status": [
    23,
    25,
    26
  ],
  "refNo": [
    "XR202201310001",
    "XR202201310002",
    "XR202101310003"
  ],
  "isAllVersion": true
}
```

- **Response HTTP Status Code:**

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

Response Body

Name	Data Type	Data Length	Description	Example
RefNo	String	20	Cross Reference Number	XR202201310001
MeasurementPeriod	DateTime		วันที่งวดข้อมูล	2022-01-31T00:00:00
VersionNo	Integer		เลขที่การประมวลผล	1
Provider.Id	String	40	รหัสองค์กร	999
Provider.Name	String	500	ชื่อองค์กร	Example Bank
Provider.NameTH	String	500	ชื่อองค์กรภาษาไทย	ธนาคารทดสอบ (จำกัด)
ValidationGroup.Id	String	10	รหัสกลุ่มการตรวจสอบ	6
ValidationGroup.Name	String	100	ชื่อกลุ่มการตรวจสอบ	DMS BLS vs RDT OTDM
ValidationGroup.Description	String	255	รายละเอียดกลุ่มการตรวจสอบ	Balance sheet (Credit) with RDT Credit
Status.Id	Integer		รหัสสถานะการตรวจสอบ	23
Status.Description	String	50	รายละเอียดสถานะการตรวจสอบ	Waiting for Related Sets
Status.Timestamp	DateTime		เวลาสถานะการตรวจสอบ	2022-02-23T14:07:42.637
Frequency.Id	String	20	รหัสความถี่การตรวจสอบ	MT
Frequency.Name	String	500	ชื่อความถี่การตรวจสอบ	Monthly

Response Body JSON Example

```
[
  {
    "RefNo": " XR202201310001",
    "MeasurementPeriod": "2022-01-31T00:00:00",
    "VersionNo": 1,
    "Provider": {
      "Id": "999",
      "Name": "Example Bank.",
      "NameTH": "ธนาคารทดสอบ จำกัด"
    },
    "ValidationGroup": {
      "Id": "6",
```

```

        "Name": "DMS BLS vs RDT OTDM ",
        "Description": "Balance sheet (Credit) with RDT Credit"
    },
    "Status": {
        "Id": 23,
        "Description": "Waiting for Related Sets",
        "Timestamp": "2022-02-23T14:07:42.637"
    },
    "Frequency": {
        "Id": "MT",
        "Name": "Monthly"
    }
}
]

```

- **Sample Code:**

```

var obj = new
{
    validationGroupId = new List<string>() { "1", "6", "7" },
    measurementPeriodFrom = new DateTime(2022, 1, 31),
    measurementPeriodTo = new DateTime(2022, 03, 31),
    processFrom = new DateTime(2022, 1, 31),
    processTo = new DateTime(2021, 4, 1),
    refNo = new List<string>() { "XR202201310001", "XR202201310002", "XR202201310003" },
    isAllVersion = true
};

var sc = new StringContent(JsonConvert.SerializeObject(obj), Encoding.UTF8, "application/json");

using var response = await client.PostAsync("https://[redacted]/partner/da-cross-validation-logs", sc);

Console.WriteLine($"Response Status {response.StatusCode}({(int)response.StatusCode})");
Console.WriteLine(await response.Content.ReadAsStringAsync());

```

10. API – Cross Validation Error Log

API สำหรับใช้ในการ Download Cross Validation Error Log ของการ Cross Validation

- HTTP Method:

HTTP Method	GET
-------------	-----

- Request Query String:

Name	Data Type	Data Length	Require	Description	Example
refNo	String	20	M	Cross Reference Number	XR202201310001

- Response HTTP Status Code:

Code	Description
200	OK (Success)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error

- Sample Code:

```
var refNo = "X002181107001";

using var response = await client.GetAsync($"https://iXXXXXXXXXXXX/partner/da-cross-validation-error-log?refNo={refNo}");

Console.WriteLine($"Response Status {response.StatusCode}{{{(int)response.StatusCode}}}");

using var fs = new FileStream($"C:\\_Temp\\{response.Content.Headers.ContentDisposition.FileName}", FileMode.Create);
using var sc = await response.Content.ReadAsStreamAsync();
sc.CopyTo(fs);
```


VI. Appendix

1. Appendix A: การเตรียมไฟล์ก่อนการส่งข้อมูล

1.1. เตรียมไฟล์ข้อมูล

1.2. เข้ารหัสไฟล์ข้อมูลที่เตรียมไว้เพื่อส่งให้ ฐปท. ด้วย Library “DAEncryptionLib.dll” ที่ฐปท. เผยแพร่บน BOT Website

<https://www.bot.or.th/Thai/Statistics/DataManagementSystem/Standard/DMSDA/Pages/DMSDA.aspx>

1.3. Parameters สำหรับการเข้ารหัส

Name	Data Type	Data Length	Require	Description	Example
InputFilePath	String	250	M	Full Path ที่อยู่ของไฟล์รวมถึงชื่อไฟล์และ Extension ไฟล์	C:\Temp\example.txt
EncryptedPath	String	250	M	Output Path ที่อยู่ของไฟล์ Output (ไม่รวมชื่อไฟล์)	C:\Temp\
x509cerForSign	X509Certificate2		M	สำหรับ Sign data (Required private key)	
x509cerForEncrypt	X509Certificate2		M	สำหรับการเข้ารหัสข้อมูล (Required public key ของ ฐปท.)	
IsSign	Boolean		M	True – สำหรับต้องการทำ Digital Signature False – ไม่ดำเนินการทำ Digital Signature	true

1.4. Download Public Key ของ ธปท. เพื่อใช้ในการเข้ารหัสไฟล์ (หากมีการเปลี่ยนแปลง Public Key ทาง ธปท. จะแจ้งล่วงหน้าให้ทราบอีกครั้ง)

1.4.1. เข้า URL

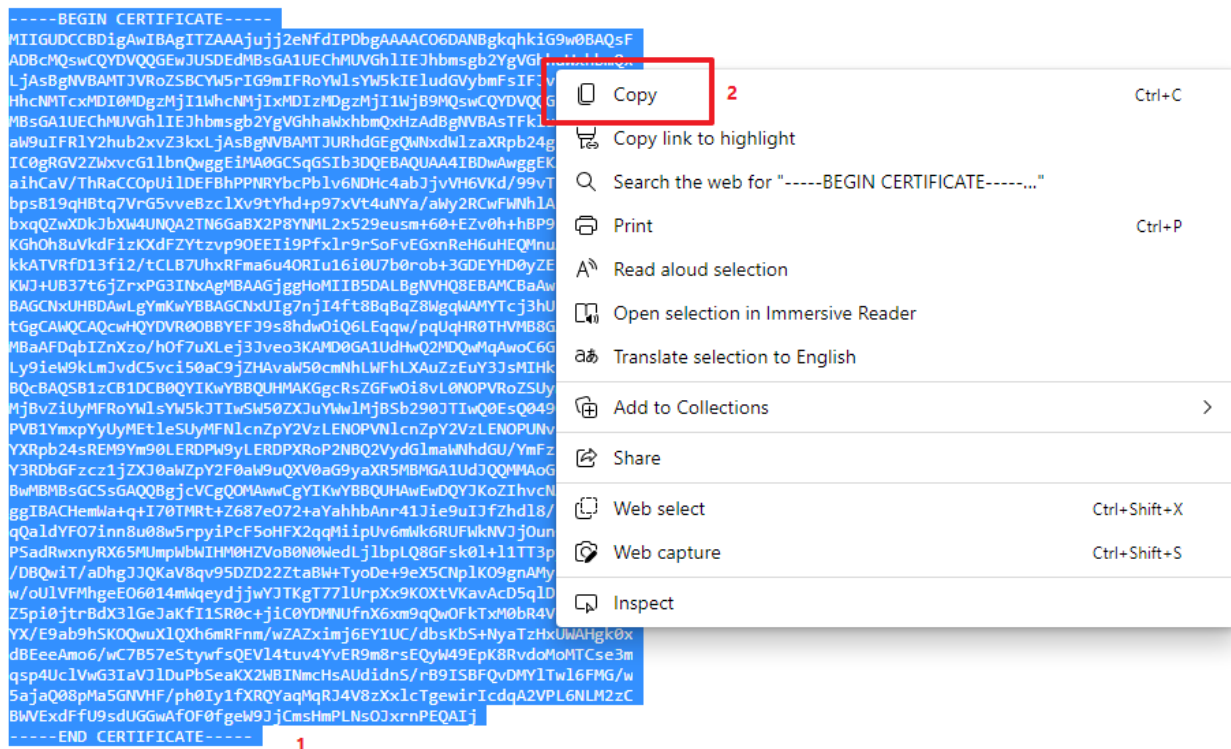
1.4.1.1. Production :

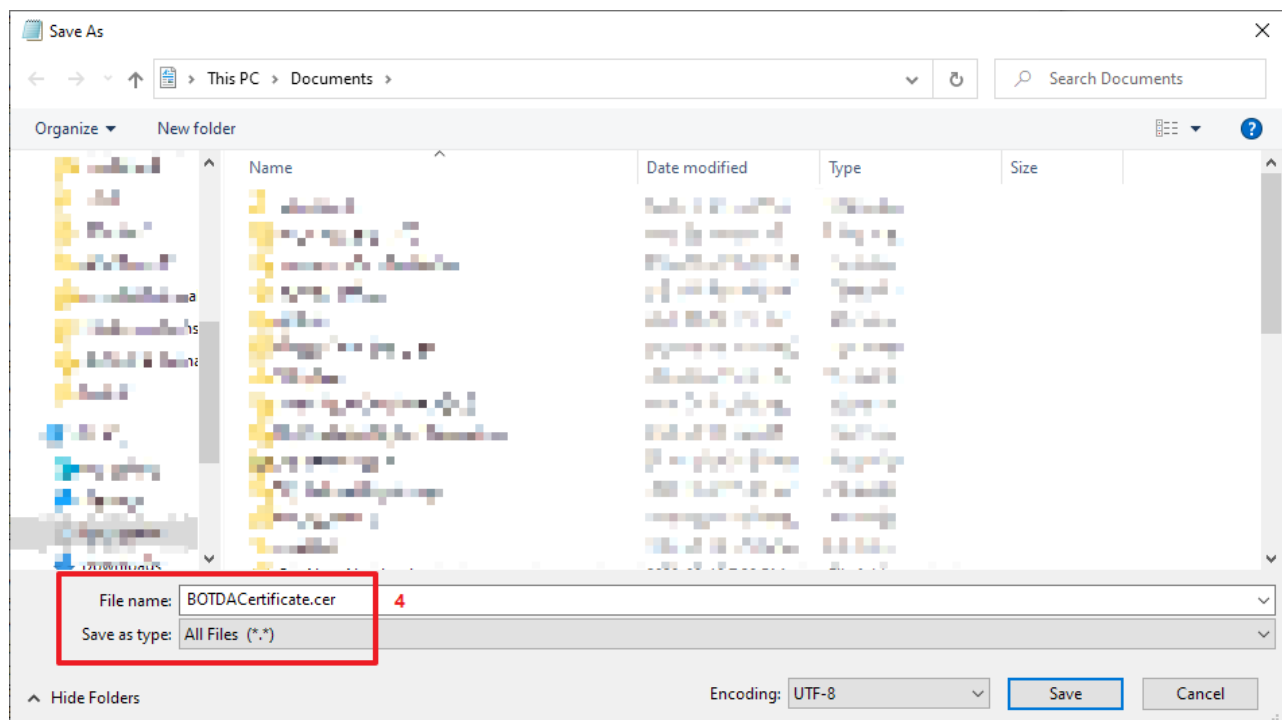
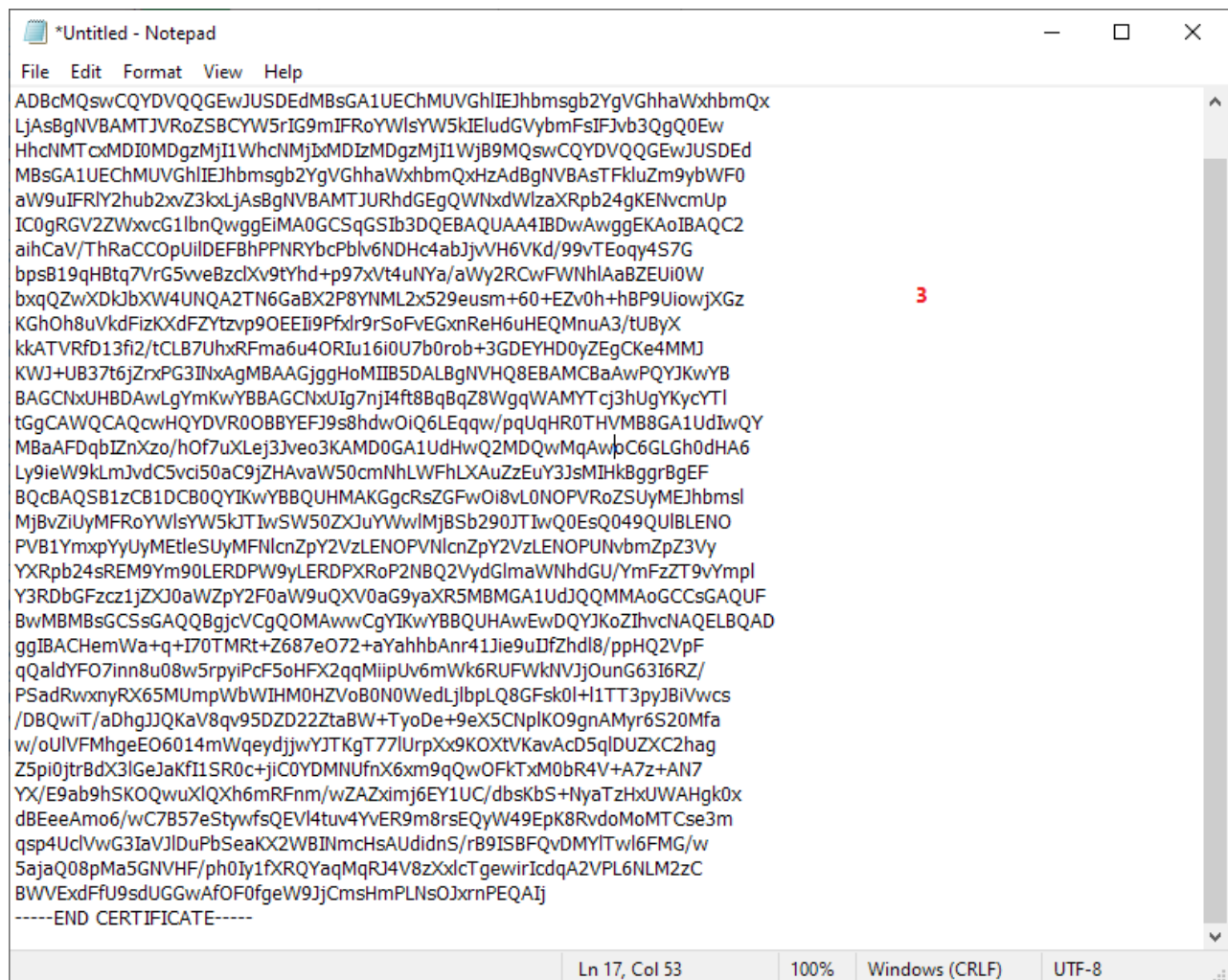
<https://www.bot.or.th/Thai/Statistics/DataManagementSystem/Standard/DMSDA/Program/botda.bot.or.th.cer.txt>

1.4.1.2. IWT :

<http://www.bot.or.th/Thai/Statistics/DataManagementSystem/Standard/DMSDA/Program/DA%20PublicKey.cer.txt>

1.4.2. Copy ข้อความทั้งหมดใน URL (1-2) ลงไปใน (3) Note Pad และ (4) Save As ชื่อไฟล์ "BOTDACertificate.cer"





1.4.3. นำ Public Key จากข้อ 1.4.2 ไปใช้ตามตัวอย่าง Sample Code ข้อ 1.5

1.5. Sample Code

```
string EncryptedPath = "C:\\outputDirectory\\";
string inputFilePath = "C:\\inputDirectory\\fileToEncrypt.doc";

string publicCerPath = "C:\\publicKey\\BOTPublicCer.cer"; // BOT Public Key use for encryption
byte[] publicCertificateStream = ReadFile(publicCerPath);
X509Certificate2 x509cerForEncrypt = new X509Certificate2(publicCertificateStream, publicCerPassword);

string privateCerPath = "C:\\privateCer\\org_cer.p12"; // Organization Private Certificate use for sign
byte[] privateCertificateStream = ReadFile(privateCerPath);
string privateCerPassword = "12345678";
X509Certificate2 x509cerForSign = new X509Certificate2(privateCertificateStream, privateCerPassword);

try
{
    string outputFile = "";
    bool isSign = true; // Required PrivateCertificate, if true, to sign on data.
    outputFile = DAEncryptionLib.DAEncryption.EncryptionProcessWrapper(inputFilePath, EncryptedPath, x509cerForSign, x509cerForEncrypt, isSign);
    Console.WriteLine("Encryption Process complete! File is located at " + outputFile);
}
catch (Exception)
{
    throw;
}
```

2. Appendix B: การส่งไฟล์ด้วยวิธี HTTP Protocol

- 2.1. ตรวจสอบ Meta Data ของการส่งไฟล์ด้วย [API – Submission Validate](#)
- 2.2. ทำการส่งข้อมูลด้วย [API – Submission](#)
- 2.3. เมื่อการส่งข้อมูลสำเร็จจะได้เลข Reference Number เพื่อใช้ในการตรวจสอบสถานะข้อมูลต่อไป

3. Appendix C: การส่งไฟล์ด้วยวิธี File Transfer Protocol

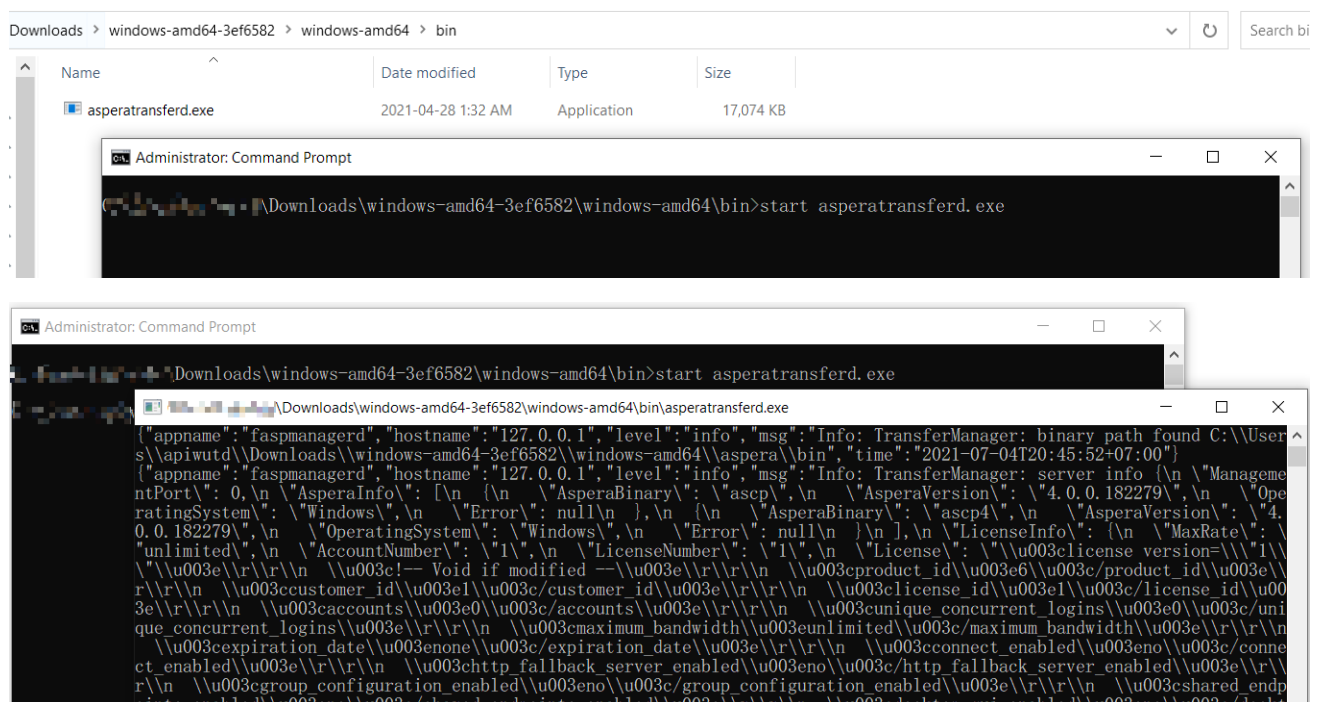
การส่งด้วยวิธีนี้จะต้องเปิด Port TCP 33001 และ UDP 33001

- 3.1. Download Aspera SDK ตาม Platform ที่พัฒนา <https://developer.ibm.com/apis/catalog/aspera--aspera-transfer-sdk/downloads/downloads.json>
- 3.2. พัฒนาระบบตามตัวอย่าง <https://developer.ibm.com/apis/catalog/aspera--aspera-transfer-sdk/Quick%20Start%20for%20Sample%20Code:%20C%20Sharp>
หรือตามตัวอย่างโปรแกรมที่ได้รับหลังจาก Download ในข้อ 3.1.

Downloads > windows-amd64-3ef6582 > windows-amd64 > connectors >				
Name	Date modified	Type	Size	
cpp	2021-04-28 1:32 AM	File folder		
csharp	2021-05-11 3:20 PM	File folder		
go	2021-04-28 1:32 AM	File folder		
java	2021-04-28 1:32 AM	File folder		
js	2021-04-28 1:32 AM	File folder		
python	2021-04-28 1:32 AM	File folder		
ruby	2021-04-28 1:32 AM	File folder		
transfer.proto	2021-04-28 1:32 AM	PROTO File	65 KB	

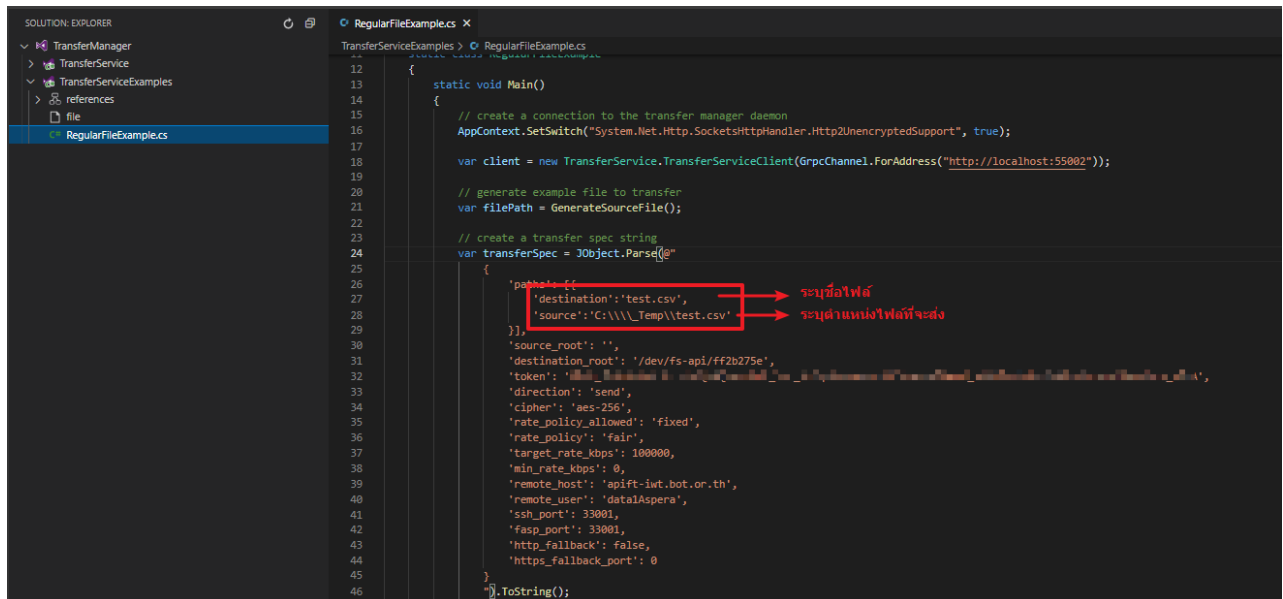
ภาพตัวอย่างโปรแกรมแต่ละภาษาที่ได้รับหลังจากการ Download ในข้อ 3.1.

3.3. Start SDK ที่ได้รับการ Download ในข้อ 3.1 ผ่าน Command Line

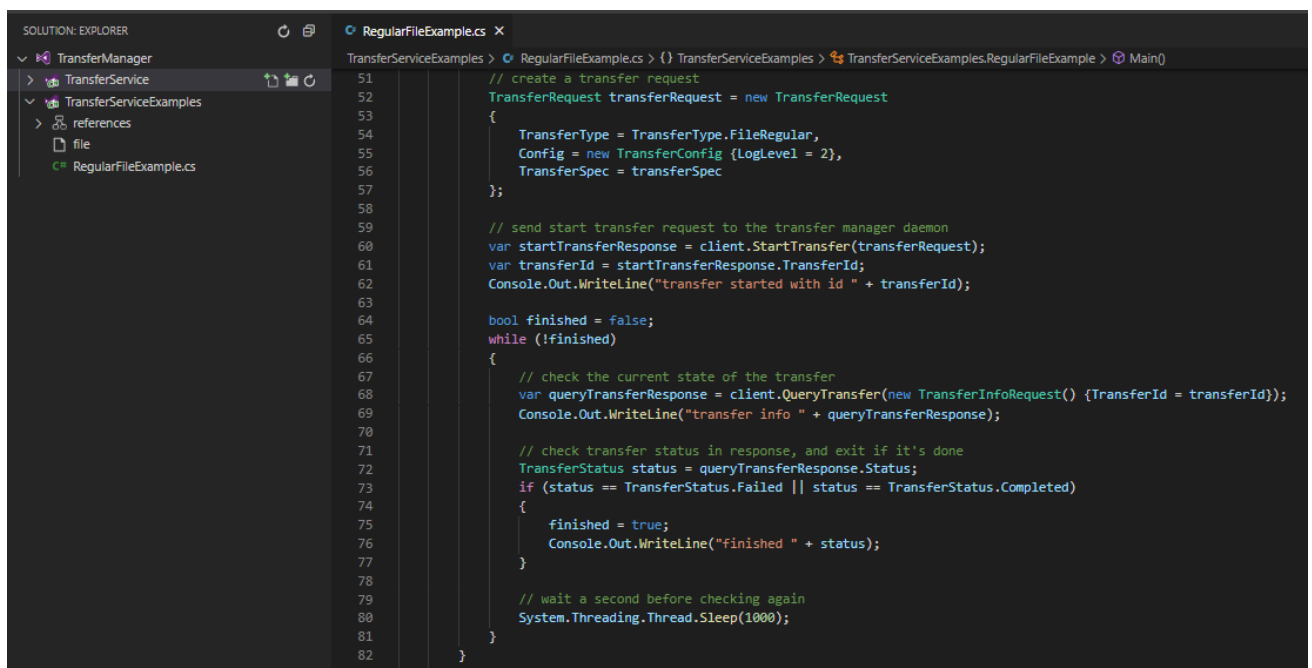


3.4. ขอ FASP Ticket จาก [API – Submission FASP Ticket](#)

3.5. นำ Ticket ที่ได้รับระบุตำแหน่งไฟล์และชื่อไฟล์ที่ต้องการส่งดังรูป (ห้ามแก้ไข Information อื่นๆ ใน Ticket)



3.6. ดำเนินการส่งไฟล์ผ่าน Aspera ด้วย Ticket ในข้อ 3.4



3.7. หลังจากส่งไฟล์เสร็จเรียบร้อยแล้วให้ดำเนินการส่ง Meta Data ของการส่งข้อมูล มายัง รพท. ผ่าน [API – Submission FASP](#)

3.7.1. ระบุ fileLocation ตาม destination_root ใน Ticket

3.7.2. ระบุ fileName ด้วยชื่อไฟล์ที่ต้องการจะส่ง

```
var transferSpec = JObject.Parse(@"
{
  'paths': [{
    'destination': '',
    'source': 'C:\_Temp\\test.csv'
  }],
  'source_root': '',
  'destination_root': '\\DataFile\\123',
  'token': '...',
  'direction': 'send',
  'cipher': 'aes-256',
  'tags': { },
  'rate_policy_allowed': 'fixed',
  'rate_policy': 'fair',
  'target_rate_kbps': 100000,
  'min_rate_kbps': 0,
  'lock_min_rate': true,
  'preserve_times': false,
  'remote_host': 'apift-iwt.bot.or.th',
  'remote_user': 'data1Aspera',
  'sshfp': null,
  'ssh_port': 33001,
  'fasp_port': 33001,
  'http_fallback': true,
  'http_fallback_port': 8443
}
").ToString();
```

3.8. เมื่อการส่งข้อมูลสำเร็จจะได้เลข Reference Number เพื่อใช้ในการตรวจสอบสถานะข้อมูลต่อไป