

No.	API Name	Description
0a	General API Structure	HTTP Request Headers, Parameters, Body and Response Body Structure
0b	Authentication/ Authorization Token Structure	JWT Token Structure
1	Submit Supporting Documents	Enables an applicant to directly submit one or many supporting documents.
2	Enquire Supporting Document Status	Request a status update on one or many supporting documents.

Operation Name		General API Structure	Description	General API Structure				
HTTP Method		GET	Description	General Request Structure for GET				
Seg No	Field Name	Description	Length	Datatype	Required (Y/N/C)	Permitted Values	Sample Value(s)	Remarks
1	HTTP Header	HTTP Header						
1.1	Authorization	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authentication .		String	Y	Request Header's Token Values	JWS eyJ2ZXI0bXlJAILCiraWQ0Oily...	
1.2	X-Trade-Finance-Token	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authorization .		String	Y	Request Header's Token Values	JWS eyJ2ZXI0bXlJAILCiraWQ0Oily...	
1.3	X-HSBC-Request-Correlation-Id	Random Universal Unique Identifier (UUID) generates in every API Request	36	String	Y	Universally Unique Identifier	Sc1fe43-cb9c-473a-a557-2f1ec6e2d145	
1.4	X-HSBC-CountryCode	The target HSBC System Location. Please check with your implementation manager.	2	String	N	ISO 3166-1 alpha-2	SG	
1.5	X-HSBC-Crypto-Signature	Optional field. Only request if client does not require message level signing.	5	Boolean	N	true false		
2	HTTP Query Parameters	Query Parameters. Refer to corresponding APIs				Query Parameters	?paramKey001=paramValue001¶mKey002=paramValue002...	
Request Query Example								
GET {{env-url}}/{{api-end-point}}?paramKey001=paramValue001¶mKey002=paramValue002...								
HTTP Method		POST	Description	General Request Structure for POST				
Seg No	Field Name	Description	Length	Datatype	Required (Y/N/C)	Permitted Values	Sample Value(s)	Remarks
1	HTTP Header	HTTP Header						
1.1	Authorization	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authentication .		String	Y	Request Header's Token Values	JWS eyJ2ZXI0bXlJAILCiraWQ0Oily...	
1.2	X-Trade-Finance-Token	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authorization .		String	Y	Request Header's Token Values	JWS eyJ2ZXI0bXlJAILCiraWQ0Oily...	
1.3	X-HSBC-Request-Correlation-Id	Random Universal Unique Identifier (UUID) generates in every API Request	36	String	Y	Universally Unique Identifier	Sc1fe43-cb9c-473a-a557-2f1ec6e2d145	
1.4	X-HSBC-CountryCode	The target HSBC System Location. Please check with your implementation manager.	2	String	N	ISO 3166-1 alpha-2	SG	
1.5	X-HSBC-Crypto-Signature	Optional field. Only request if client does not require message level signing.	5	Boolean	N	true false		
1.6	X-HSBC-Request-Idempotency-Key	The unique identifier (UUID) that you submit as a request header, that helps bank to identify your request is not duplicated.	36	String	Y	Universally Unique Identifier	Sc1fe43-cb9c-473a-a557-2f1ec6e2d145	
2	HTTP Body	Encrypted Request Body to be sent to bank, Base 64 Encoded.		application/json	Y	Stringified JSON Object		
2.1	encryptedRequestBase64	Base64 Encoded string of the encrypted message. Please encrypt it with the bank's public key.		String	Y	Base64 String	LS0tLS1CRUdJTiBQR1AgTUUVU0FHR...	
HTTP Request Body Example								
{"encryptedRequestBase64":"LS0tLS1CRUdJTiBQR1AgTUUVU0FHR..."}								
HTTP Method		GET/ POST Response	Description	General Response Structure				
Message Type		Encrypted Response Body						
Seg No	Field Name	Description	Length	Datatype	Required (Y/N/C)	Permitted Values	Sample Value(s)	Remarks
1	HTTP Status	Status of the HTTP Request.			Y	200: SUCCESS (Success Response from Get Request) 201: Created (Success Response from Post Request) 400: Bad Request Headers/ Payload 401: Authentication/ Authorization Failure 403: Entitlement Validation Failure 422: Crypto Failure 500: Service Failure		
2	HTTP Body	Encrypted Response Body (by client's pgp key) received from bank, Base 64 Encoded		application/json	Y	Stringified JSON Object		
2.1	encryptedResponseBase64	Base64 Encoded string of the encrypted message. Please encrypt it with the client's public key.		String	Y	Base64 String	LS0tLS1CRUdJTiBQR1AgTUUVU0FHR...	
HTTP Response Body Example								
{"encryptedResponseBase64":"LS0tLS1CRUdJTiBQR1AgTUUVU0FHR..."}								
Message Type		Error Messages						
Seg No	Field Name	Description	Length	Datatype	Required (Y/N/C)	Permitted values	Sample Value(s)	Remarks
1	type	Type of error	255	String	Y		/problem-details/types/validation-errors	
2	title	Title of error	255	String	Y		Fields invalid	
3	status	HTTP Status Code	225	String	Y		400	
4	detail	Detail description of error	225	String	Y		Fields invalid	
5	errors	Multiple Errors		Array	N			
5.1	type	Type of error	225	String	Y		/problem-details/types/validation-errors	
5.2	detail	Title of error	225	String	Y		data.invoices[0].externalInvoiceId must not be null.	
6	instance	X-HSBC-Request-Correlation-Id of the corresponding request	36	String	Y	Universally Unique Identifier	Sc1fe43-cb9c-473a-a557-2f1ec6e2d145	
7	errorDateTime	Datetime of error	225	String	Y		2024-08-21T12:20:46.655Z	
Single Error Example								
<pre>{ "type": "/problem-details/types/validation-errors", "title": "Fields invalid", "status": "400", "detail": "data.invoices[0].externalInvoiceId must not be null.", "instance": "ae7e19bc-a9dd-447b-b230-cd8712576417", "errorDateTime": "2023-08-27T21:17:09.466Z" }</pre>								
Multiple Errors Example								
<pre>{ "type": "/problem-details/types/validation-errors", "title": "Fields invalid", "status": "400", "detail": "Fields invalid", "errors": [{ "type": "/problem-details/types/validation-errors", "detail": "data.invoices[0].externalInvoiceId must not be null." }, { "type": "/problem-details/types/validation-errors", "detail": "data.invoices[0].sellerId must not be null." }] "instance": "86838151-4acb-42ae-a81c-4edc0648d3b5", "errorDateTime": "2023-08-27T21:17:45.710Z" }</pre>								
Errors and Troubleshooting								
type	title	status	detail/ what is the error	Troubleshooting	Remarks			
/problem-details/types/validation-errors#paramKey001%	Fields invalid	400	Key and value must not be blank. should be no longer than %maxLength% characters. must be %enum3%;%enum2%;%enum3%. must be provided in format of YYYY-MM-DD and correct date time. must match the pattern of [0-9]{1,14}[\V{0-9}]{1,3}?	These are requestor side's errors related to invalid input of Request Header/ Query Parameters/ Unencrypted JSON Body. Please refer to corresponding API Specification and correct the errors.				

/authn-error/code/EDSPER1001			Mandatory field kid is missing or invalid in the customer JWT header	Check if the correct key ID (Client's private key) in the token.	Request Header's Token Values
/authn-error/code/EDSPER1002			Invalid PGP Key , Key is set to never expiry	Contact client implementation manager to re-submit a new client public key with expiration	
/authn-error/code/EDSPER1003			Invalid PGP Key , Key is expired	Check if the correct Client's Private key is used to sign the token.	
/authn-error/code/EDSPER1004			Invalid PGP Key	Check if "iat" in the token is generated based on current time.	
/authn-error/code/EDSPER1005			Customer JWT is expired	Check if the valid signature algorithm is inputted in the token.	
/authn-error/code/EDSPER1006			Mandatory field claim "iat" is missing or invalid from customer JWT		
/authn-error/code/EDSPER1007			Mandatory field claim "Alg" is missing or invalid in customer JWT header		
/authn-error/code/EHVDERS010			Algorithm is Not Supported		
/authn-error/code/EDSPER1008			Mandatory field Sub (profile id) is missing or invalid in customer JWT body	Check if "sub" is inputted in the token claims.	Request Header's Token Values
/authn-error/code/EDSPER2002			User does not exist	Check if a valid "sub" is inputted in the token claims. This value is the profile ID you received from the client implementation manager	
/authn-error/code/EDSPER2003			Mandatory field jti is missing or invalid in customer JWT payload	Check if a "jti" is a valid uuid	
/authn-error/code/EDSPER2004			User account is locked	Please wait for 5 minutes and retry.	
/authn-error/code/EDSPER2005			JWT Signature validation failed / Invalid JWT Signature	Check if the correct Client's Private key is used to sign the token.	
/authn-error/code/EDSPER2006			Blocked access to user accessing from sanctioned country		
/authn-error/code/EDSPER2007			IP address missing so sanctioned country check failed.		
/authn-error/code/EDSPER2008			Invalid Authentication Token	Check if Token is inputted in the Request Header: Authorization: JWS ({token}) X-HSBC-Trade-Finance-Token: JWS ({token})	Request Header's Token Values
/authn-error/code/EHVDERS008			Either Client Token or Payload is in invalid format	Check if the HTTP request body format is correct. Refer to General API Request Structure	
/authn-error/code/EDSPER2009			Mandatory field aud is missing or invalid in customer JWT payload	Check if "aud" is inputted in the token claims as "baas".	Request Header's Token Values
/authn-error/code/EHVDERS009			Mandatory field ver is missing or invalid from JWT header	Check if "ver" is inputted in the token headers as "1.0"	
/authn-error/code/EDSPER2010			User Login credentials are invalid	Please contact Client Implementation Manager	
/authn-error/code/EDSPER2012			Partner-customer relationship verification failed	Check if "obo.sub" (Customer Profile ID) in the token claims is correct and entitled to access the corresponding api endpoint	Request Header's Token Values
/authn-error/code/EDSPER2013			Payload hash in the token claims does not match the HTTP Request Body	Check if the Payload Hash is generated from the HTTP Request Body.	
/authn-error/code/EDSPER2014					
/authn-error/code/EHVDERS011					
/problem-details/types/customer-info-invalid	Customer info invalid	403	AuthZ check failed, please check your customer info	Check if valid "sub", "obo.sub" in the token claims is inputted along with a valid auth-z identifier. The Auth-z Identifier varies for different API. Please refer to the corresponding specification.	
/problem-details/types/data-not-found		404	data not found	Try to widen the search criteria (Query Parameters in the Get Request)	
/problem-details/types/incorrect-idempotency-key	Header Required Field Incorrect	409	Duplicated X-HSBC-Request-Idempotency-Key: #%idempotencykeyValue%	Check if the request has already been sent. For a new request, use a new uuid.	
PAYLOAD-INCORRECT	Request Payload Incorrect	422	rovided Key id does not match the key client used on this payload	Check if the correct bank public key is used to encrypt the request.	
/problem-details/types/internal-server-error	Internal Server Error	500	Internal Server Error - This includes uncategorized Exceptions	Contact Client Implementation Manager	
/problem-details/types/bad-gateway	Bad Gateway	502			
/problem-details/types/service-unavailable	Service Unavailable	503	Bank System Outage	Retry in a while. Contact Technical Support if Problem persist.	
/problem-details/types/gateway-timeout	Gateway Timeout	504			

Operation Name	Authentication/ Authorization Token	Description	API Authentication is performed using a signed JSON Web Token (JWT). The token must be provided in the header of the HTTP request every time you invoke HSBC's API services. This ensures only authorised and pre-approved organisations can access and utilise our API services.
Actor	Client/ Partner's System		The JWT is generated by the partner, not provided by HSBC. There are many utilities available that can provide JWT generation and validation logic for all commonly used programming languages. The following example uses the "jsonwebtoken" java library.
Reference	How to generate JSON Web Token	Pre-requisite	<ol style="list-style-type: none"> Client/ Partner Private Encryption Key - for signing the JSON Web Token your system generated. Client/ Partner Public Encryption Key alias - The Public Key should be shared with HSBC as part of the client/ partner on-boarding process. Client/ Partner Profile ID - The identifier HSBC provided you during the on-boarding process. Customer Profile ID - The identifier used to identify the customer of your platform.

Seq. #	Field Name	Description	Length	Data Type	Required (Y/N)	Permitted values	Sample Value(s)	Remark
1	HTTP Header	HTTP Header						
1.1	Authorization	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authentication .		String	Y	Serialized Compact of JSON Web Token	JWS eyJ2ZXI0bXlJAILCiraWQIOily...	
1.1.1	header	Header of the token.		Object	Y			
1.1.1.1	typ	Token Type. Must be set to "JWT".	3	String	Y	"JWT"	"JWT"	
1.1.1.2	kid	Public Key Id which HSBC uses to verify JWT signature. HSBC reads the public key by this key ID, from the PGP key file provided by our client/ partners, to verify JWT signature.	16	String	Y	Long.toHex(encryptionKeyId)	"4DBDAA833163AEC7"	{ "typ": "JWT", "kid": "4DBDAA833163AEC7", "alg": "RS256", "ver": "1.0" }
1.1.1.3	alg	Algorithm to verify the signature on the token. The following algorithms are supported: RS256, RS384, RS512, PS256, PS384, PS512.	5	String	Y	RS256, RS384, RS512, PS256, PS384, PS512	"RS256"	
1.1.1.4	ver	HSBC private claim. Currently the partner MUST set it as "1.0".	3	String	Y	1.0	"1.0"	
1.1.2	claims	Claims of the token			Y			
1.1.2.1	jti	Case sensitive unique identifier of the token even among different issuers. There is a negligible probability that the same value will be accidentally assigned to a different data object. If the application uses multiple issuers, collisions MUST be prevented among values produced by different issuers as well. The "jti" claim can be used to prevent the JWT from being replayed. The "jti" value is a case-sensitive string.	36	String	Y	Universally Unique Identifier	"6049b9b2-e82d-467e-ac5c-fcc595be9ca5"	{ "jti": "6049b9b2-e82d-467e-ac5c-fcc595be9ca5", "sub": "TAAS00000", "aud": "baas", "iat": 1717562115, "obo": { "sub": "customer001" }, "payload_hash_alg": "RSASHA256", "payload_hash": "3cd2a2b75fd4f8cb..." }
1.1.2.2	iat	Identifies the time at which the JWT was issued. The value must be a NumericDate. Please note, this time is generated by your system, hence please make sure your system is synchronized with Internet Clock.	10	Long	Y	Current Time in milliseconds / 1000	1717562115	
1.1.2.3	sub	Identifies the subject of the JWT. This would be the Client/ Partner API Profile provided from HSBC.		String	Y		"TAAS00000"	
1.1.2.4	obo	On-behalf Of. Only required if you are an aggregation platform		Object	N			
1.1.2.4.1	sub	Customer Profile ID. Unique Identifier you assigned for your platform's customer.	255	String	Y		"customer001"	
1.1.2.5	aud	Identifies the recipients that the JWT is intended for. Always "baas".	4	String	Y	"baas"	"baas"	
1.1.2.6	payload_hash	Hashing value of the HTTP request payload body. This is used for Mule policy to do validation		String	Y	String of Message Digest	"3cd2a2b75fd4f8cb..."	
1.1.2.7	payload_hash_alg	Hashing algorithm of the payload_hash.		String	Y	RSASHA256, RSASHA384, RSASHA512	"RSASHA256"	
1.2	X-Trade-Finance-Token	A digital signature (token) consists of header and claims, generate by Client/ Partner in every API Request for Authorization .	255	String	Y	Serialized Compact of JSON Web Token	JWS eyJ2ZXI0bXlJAILCiraWQIOily...	
1.2.1	header					Same as 1.1.1		
1.2.2	claims					Same as 1.1.2		
1.2.2.1	jti					Same as 1.1.2.1		
1.2.2.2	iat					Same as 1.1.2.2		
1.2.2.3	sub					Same as 1.1.2.3		
1.2.2.4	obo					Same as 1.1.2.4		
1.2.2.4.1	sub					Same as 1.1.2.4.1		{ "jti": "6049b9b2-e82d-467e-ac5c-fcc595be9ca5", "sub": "TAAS00000", "aud": "baas", "iat": 1717562115, "obo": { "sub": "customer001" }, "accountNumber": "SGHSBC123456789012", "productType": "TradeLoan", "checksum": "huzd..." }
1.2.2.5	aud					Same as 1.1.2.5		
1.2.2.6	accountNumber	Account number for the client, such as trade account for loan product, client identifier account for RF product, merchantId for eMarketFinance	255	String	Y		SGHSBC123456789012	
1.2.2.6.1	productType	Add different type to client to mark target upload file usage. Core Trade uses TradeLoan, DocumentaryCredit, Bill, GuaranteeStandbyDc, SupplyChainFinance. Receivable Finance uses ReceivableFinance. eMarketFinance uses EMarketFinanceOnboard, EMarketFinanceMonitor. EMarketFinanceOnboard is used for onboard data, EMarketFinanceMonitor is used for monitor data.	255	String	Y	Enum: TradeLoan DocumentaryCredit Bill GuaranteeStandbyDc SupplyChainFinance ReceivableFinance EMarketFinanceOnboard EMarketFinanceMonitor Other		
1.2.2.6.2	checksum	Checksum of attachment document.	255	String	Y			

Operation Name	Enquire Export DC's Status	Description	Enables an applicant to directly submit one or many supporting documents.
Service Interface	/v3/filemgr/supporting-files	Pre-requisite	
HTTP Method	POST	Post condition	
Authorization ID	Customer's Account Number. This value is provided in the JSON Web Token (jwt.claims.accountNumber). Please refer to Request Header's Token		Request Header's Token

Message Type		Query Parameters						
Seg lvl	Field Name	Description	Length	Data Type	Required (Y/N)	Permitted values	Sample Value(s)	Relationship
2.1.2	attachDocument	Document the client want to upload to HSBC. Multipart/form-data: Multipart/form-data allows you to submit binary data and multiple data types in a single request. For example, you can use these requests for file uploads and transferring a file with a JSON object.	20	multipart/form-data	N			

Message Type		Decrypted Response						
Seg lvl	Field Name	Description	Length	Data Type	Required (Y/N)	Permitted values	Sample Value(s)	Relationship
1	data			Object	Y			
1.1	docUploadRequestid			String	N			
1.2	accountNumber	Account number for the client, such as trade account for loan product, client identifier account for RF product, merchantid for eMarketFinance.		String	N			
1.3	productType	Add different type to client to mark target upload file usage. Core Trade uses TradeLoan, DocumentaryCredit, Bill, GuaranteeStandbyDc, SupplyChainFinance. Receivable Finance uses ReceivableFinance. eMarketFinance uses EMarketFinanceOnboard, EMarketFinanceMonitor. EMarketFinanceOnboard is used for onboard data, EMarketFinanceMonitor is used for monitor data.	2	String	N	Enum: TradeLoan DocumentaryCredit Bill GuaranteeStandbyDc SupplyChainFinance ReceivableFinance EMarketFinanceOnboard EMarketFinanceMonitor Other	TradeLoan	

JSON Schema Example - Decrypted

```
{
  "data": {
    "docUploadRequestid": "84f21a7bd88b45218a6a228b7f16205d",
    "accountNumber": "5GHSBC123456789012",
    "productType": "TradeLoan"
  }
}
```

Operation Name	Enquire Supporting Documents	Description	Request a status update on one or many supporting documents.
Service Interface	/v3/filemgr/supporting-files	Pre-requisite	
HTTP Method	GET	Post condition	
Authorization ID	Customer's Account Number. This value is provided in the JSON Web Token (jwt.claims.accountNumber). Please refer to Request Header's Token		Request Header's Token

Message Type		Query Parameters						
Seq No	Field Name	Description	Length	Data Type	Required (Y/N)	Permitted values	Sample Value(s)	Relationship
2.1	docUploadRequestId	Request id for document upload, generated by HSBC API, client will use this request id to get the status of the document, uuid	255	String	C		84f21a7bd88b45218a6a228b7f16205d	of the fileUploadRequestId and docId must be input
2.2	docId	Document id, returned by File Net. Doc id for the file upload to HSBC	255	Array of Strings	C		doc1	
Request Query Example								
Enquire All Documents		GET {{[env-url]}/v3/filemgr/supporting-files/docUploadRequestId=84f21a7bd88b45218a6a228b7f16205d}						
Enquire specific Documents		GET {{[env-url]}/v3/filemgr/supporting-files/docId={doc1,doc2}}						

Message Type		Decrypted Response						
Seq No	Field Name	Description	Length	Data Type	Required (Y/N)	Permitted values	Sample Value(s)	Relationship
1	data			Object	Y			
1.1	docUploadRequestId	Request id for document upload, generated by HSBC API, client will use this request id to get the status of the document	255	String	Y	uuid	84f21a7bd88b45218a6a228b7f16205d	
1.2	docDetails				Y			
1.2.1	formType	Request form type. Has option P (portal documents). It means data upload by portal. Returned by HTS downstream API when upload document to FileNet. Has options EMarketFinanceOnboard or EMarketFinanceMonitor when upload document to DB of EMarketFinance product.	255	String	Y	Enum: P EMarketFinanceOnboard EMarketFinanceMonitor	P	
1.2.2	id	Document id, returned by HSBC FileNet. Doc id for the file upload to HSBC	255	String	Y	uuid	doc1	
1.2.3	name	Document Name the document uploaded by the client	255	String	Y		file1.txt	
1.2.4	numPages	Number of the document page	255	String	Y		1	
1.2.5	imageFormat	Document format for the uploaded document	255	String	Y		txt	
1.2.6	imageBytes	Document size within Bytes	255	String	Y		20967854	
1.2.7	hash	Hash value for the uploaded document security check, returned by HTS downstream API.	255	String	Y		xyz...	
1.2.8	docStatus	DC status		Object	Y			
1.2.8.1	value	Document upload status when upload the documents to HSBC FileNet API. Status include Pending, Scanning, ScanFailed, Parsing, Failed, Uploaded/Success. For the uploaded document with Virus Scan failed, cannot search out.	10	String	Y	Enum: Pending Scanning ScanFailed Parsing Failed Uploaded/Success	Success	
1.2.9	failedReason		255	Object	N			
1.2.9.1	failedCode	Failed code. It will show when document upload failed.	255	String	N	"E000:Capture a Non-MKT error!"; "E001:Virus Scan failed."; "E002:Virus Scan timeout."; "E003:Merchant Id is not matched."; "E004:%s is missing or value empty."; "E005:%s length or precision is wrong, value is %s."; "E006:No data found in Data file."; "E007:Value %s does not match header type."; "E008:Number of data fields does not match number of headers."; "E009:Data file parsing failed."; "E010:Data file parsing timeout."; "E011:Upload file failed."; "E012:Upload file timeout.";	E001	
1.2.9.2	failedMessage	Message to describe the reason why the uploaded document upload failed. It will show when document upload failed.	255	String	N	"E000:Capture a Non-MKT error!"; "E001:Virus Scan failed."; "E002:Virus Scan timeout."; "E003:Merchant Id is not matched."; "E004:%s is missing or value empty."; "E005:%s length or precision is wrong, value is %s."; "E006:No data found in Data file."; "E007:Value %s does not match header type."; "E008:Number of data fields does not match number of headers."; "E009:Data file parsing failed."; "E010:Data file parsing timeout."; "E011:Upload file failed."; "E012:Upload file timeout.";	Virus Scan failed	

JSON Schema Example - Decrypted

```
{
  "data": {
    "docUploadRequestId": "84f21a7bd88b45218a6a228b7f16205d"
  },
  "docDetails": [
    {
      "formType": "P",
      "id": "84f21a7bd88b45218a6a228b7f16205d",
      "name": "file1.txt",
      "numPages": "1",
      "imageFormat": "txt",
      "imageByte": "20967854",
      "hash": "xtz...",
      "docStatus": {
        "value": "Success"
      }
    }
  ]
}
```