

C2B INTEGRATION REST API v1.0 specification

**Payment System
for International Bank of Azerbaijan**

Document information

Project name: Open API for Transfers
Document name: C2B_INTEGRATION REST API v1.0 specification

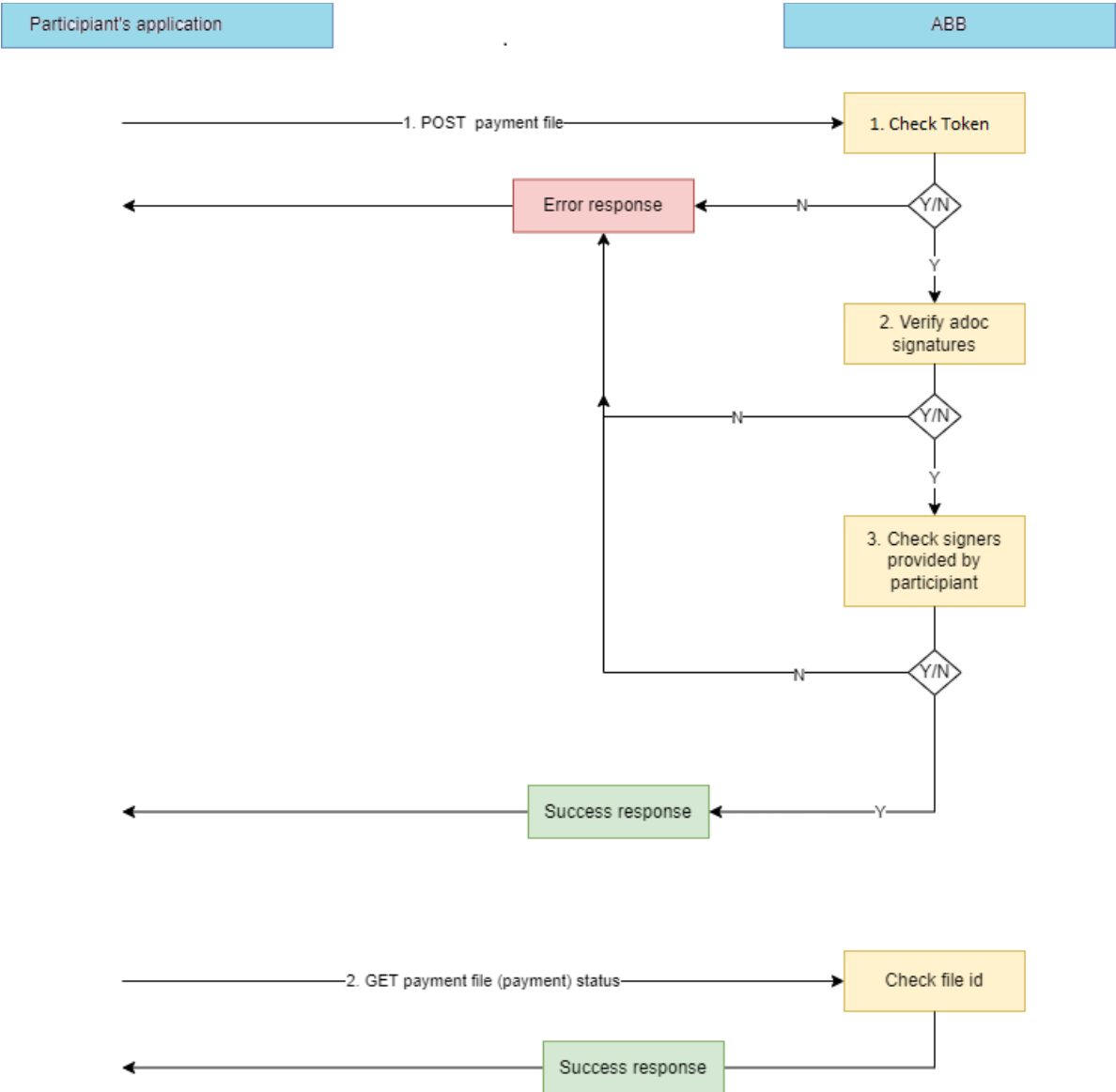
Version history (in the reversed order)

Version	Date	Authors	Comments
001	2022-10-19	CIT Corporate Integration Squad	First version
002	2023-04-19	CCI Corporate Customer Integration	Second version
003	2023-11-02	CCI Corporate Customer Integration	Third version
004	2024-03-01	CCI Corporate Customer Integration	Fourth version

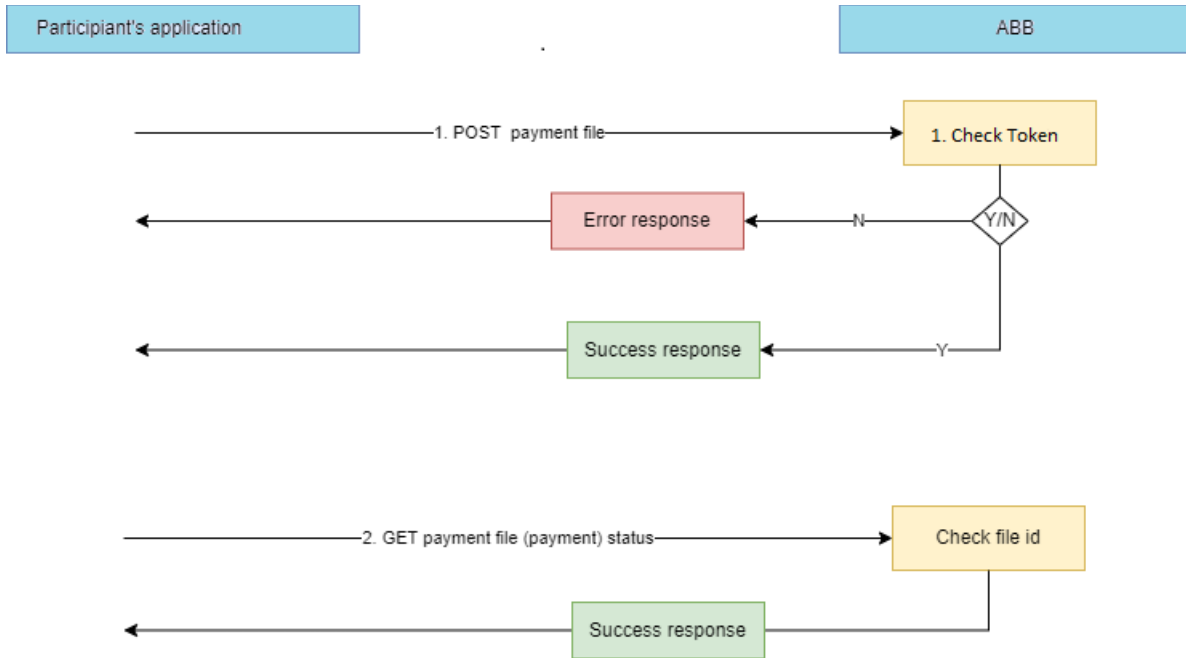
Table of contents

1.	Process flow signed	4
2.	Process flow regular payments	5
3.	Business process actions	6
3.1.	POST/payments/signed	6
3.2.	POST/payments/.....	7
3.3.	POST/payments/otp.....	7
3.4.	POST/payments/verify-otp.....	7
3.5.	GET/payments/	8
3.6.	File status	8
3.7.	Payment status	8
4.	Methods definition and format	9
4.1.	Generate token	9
4.2.	Post signed payments.....	11
4.3.	Post payments	13
4.4.	Post payments with otp.....	15
4.5.	Post payments verify-otp	17
4.6.	Get all payments	19
4.7.	Get individual payments	21
4.8.	Get balance.....	23
4.9.	Get Account Statement.....	25
4.10.	Post Salary Payments.....	28
4.11.	Post signed Salary Payments.....	30
4.12.	Get all Salary Payments.....	32
5.	XML structure	34
5.1.	Payment Order XML Structure.....	34
5.2.	Explanation of XML structure	36
5.3.	Salary Payment XML Structure.....	37

1. Process Flow (signed)



2. Process Flow (regular payments)



3. Business process actions in the diagram above (process flow for “signed” and “regular” payments) :

- 3.1. POST/payments/signed** – Participant’s application (client) sends request to generate payments. The payment file is signed with “ASAN IMZA” and it is posted by the service in “BASE64” format. In addition to the file the “External file ID” field can also be sent in the service request. This is meant as a unique ID for the file. The bank accepts the request and checks.
- **Check authorization** – The “Bearer Token” that’s provided by the bank to the client will be checked. If it is false then the response will return an “unauthorized (401) ” error message. Otherwise, in true case it will go to the next step.
 - **Verify adoc signatures** – The adoc file is opened and the authenticity of the signatures will be checked with “ASAN IMZA”. If it is false then the response will return corresponding error , otherwise it will go to the next step.
 - **Check signers provided by participant** – The FIN of signed users will be compared with the FIN of signature authorized persons given to the bank by the client. If it is false then the response will return corresponding error , otherwise will return “success response”. Generated unique ID will return in this response.
 - **Success response** - The transaction execution process will begin after successful verification of the posted data in accordance with the bank’s internal rules. Result information will be obtained with the GET method.

3.2. POST/payments/ – Participant's application (client) sends request to generate payments. The payment file will be posted by the service in "BASE64" format. In addition to the file the "External file ID" field can also be sent in the service request. This is meant as a unique ID for the file. The bank accepts the request and checks.

- **Check authorization** – The "Bearer Token" that's provided by the bank to the client will be checked. If it is false then the response will return an "unauthorized (401)" error message. Otherwise, in true case it will go to the next step.
- **Success response** – The transaction execution process will begin after successful verification of the posted data in accordance with the bank's internal rules. Result information will be obtained with the GET method.

3.3. POST/payments/otp – Participant's application (client) sends request to generate payments. The payment file will be posted by the service in "BASE64" format. In addition to the file the "External file ID" field can also be sent in the service request. This is meant as a unique ID for the file. The bank accepts the request and checks.

- **Check authorization** – The "Bearer Token" that's provided by the bank to the client will be checked. If it is false then the response will return an "unauthorized (401)" error message. Otherwise, in true case it will go to the next step.
- **Success response** – When a batch is a created (batch returned in service response) , an OTP code is sent by the bank to the mobile number provided to the bank beforehand.

3.4. POST/payments/verify-otp – The OTP code must be resent to the bank with the batch number. If any of this information is false then the response will return corresponding error , otherwise will return "success response".

- **Success response** – The transaction execution process will begin after successful verification of the posted data in accordance with the bank's internal rules. Result information will be obtained with the GET method.

3.5. GET/payments/ - The customer sends a request to find out the status of the payment file. We can find out both the status of the file and the status of all payments on the file with this request. Also, by sending the payment ID only the status of that payment can be found.

3.5.1. GET/payments/{batchNumber} – Result of file details and payment status will be found out by sending batch number

3.5.2. GET/payments/{batchnumber}/paymentId – Result of payments will be found out individually by sending payment ID.

3.6. The possible file statuses are shown below:

- **COMPLETED** – All payments on the file have completed.
- **FAILURE** - All payments on the file have failed.
- **IN_PROGRESS** – All or several payments are in progress.
- **PARTIAL** – There are both successful and failed payments on the file.
- **ERROR** – The request of the POST method is not correct.

3.7. The possible payment statuses are shown below:

- **SUCCESS**
- **IN_PROGRESS**
- **ERROR**
- **FAILURE**

4. Methods definition and format

4.1. Generate Token

Description	Request to generate token			
Method	GET/payments/auth/token			
URL	https://api-test-c2b.abb-bank.az/payments/auth/token			
Request headers	Content-type	application/json		
	Charset	UTF-8		
Query parameters	Field	Type	Required	Description
	username	String	True	username
	password	String	True	password
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXZW50IiwiaWF0Ijoi 1487148714", "expires_in": 8640000, "refresh_expires_in": 8640000, "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXZW50IiwiaWF0Ijoi 1487148714", "token_type": "Bearer", "not-before-policy": 0, "session_state": "864e6a95-521e-4037-a892- 5584c5ff5baf", "scope": "email profile" }</pre>			

Error response	<pre>{ "timestamp": "2023-11-02T12:11:22.550+00:00", "status": 500, "error": "Internal Server Error", "message": "", "path": "/c2b-payments-ms/payments/auth/token" }</pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.2.POST signed payments

Description	Request to send "signed" payments			
Method	POST/payments/signed			
URL	https://api-test-c2b.abb-bank.az/payments/signed			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre>{ "base64aDoc": "string", "externalReference": "string" }</pre>			
Request parameters	Field	Type	Required	Description
	base64aDoc	String	True	Signed adoc file
	externalReference	String	False	Unique ID
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "data": { "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57" } }</pre>			
Error response	<pre>{ "statusCode": "ERR2", "statusDesc": "File is not in base64 format", "timeStamp": "2022-11-30 10:39:16" }</pre>			

	}			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.3.POST payments

Description	Request to send payments			
Method	POST/payments/			
URL	https://api-test-c2b.abb-bank.az/payments/			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre>{ "base64aDoc": "string", "externalReference": "string" }</pre>			
Request parameters	Field	Type	Required	Description
	base64aDoc	String	True	File in "BASE64" format
	externalReference	String	False	Unique ID
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "data": { "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57" } }</pre>			
Error response	<pre>{ "statusCode": "ERR2", "statusDesc": "File is not in base64 format", "timeStamp": "2022-11-30 10:39:16" }</pre>			

	}			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.4.POST payments with OTP

Description	Request to send payments with OTP			
Method	POST/payments/otp			
URL	https://api-test-c2b.abb-bank.az/payments/otp			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre> { "base64aDoc": "string", "externalReference": "string" } </pre>			
Request parameters	Field	Type	Required	Description
	base64aDoc	String	True	Signed adoc file
	externalReference	String	False	Unique ID
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre> { "data": { "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57" } } </pre>			
Error response	<pre> { "statusCode": "ERR2", "statusDesc": "File is not in base64 format", } </pre>			

	<pre>"timeStamp": "2022-11-30 10:39:16" }</pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.5.POST payments verify-otp

Description	Request to verify payments with otp			
Method	POST/payments/verify-otp			
URL	https://api-test-c2b.abb-bank.az/payments/verify-otp			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre> { "batchNumber": "string", "otpCode": "string" } </pre>			
Request parameters	Field	Type	Required	Description
	batchNumber	String	True	File in "BASE64" format
	otpCode	String	True	Otp code
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre> { "status": "Successfully authorized" } </pre>			
Error response	<pre> { "statusCode": "BAD_REQUEST", "statusDesc": "OTP code is wrong", "timeStamp": "2024-03-27 09:59:30" } </pre>			

Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.6.GET all payments

Description	Request to get payment and file status			
Method	GET/payments/{batchNumber}			
URL	https://api-test-c2b.abb-bank.az/payments/{batchNumber}			
Request headers	Content-type		application/json	
	Charset		UTF-8	
	Authorization		Bearer {token}	
Request parameters	Field	Type	Required	Description
	batchNumber	String	True	Generated unique ID of file
Response type	200		Success response	
	400		Bad Request	
	500		Internal Server Error	
Success response	<pre>{ "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57", "status": { "status": "IN_PROGRESS", "description": "IN_PROGRESS" }, "payments": [{ "paymentId": "2289e221-b0df-4985-8c4e-8c6c8a31cc72", "transactionReference": "120INTF220016506", "paymentStatus": { "status": "IN_PROGRESS", "transactionReference": "120INTF220016506" }, "paymentAmount": 22, "recipientAccount": "/AZ66IBAZ40140018409327445353",</pre>			

	<pre> "paymentTime": "2022-11-30 10:40:05" }] } </pre>			
Error response	<pre> { "statusCode": "ERR5", "statusDesc": "File with this batch number not found", "timeStamp": "2022-11-30 10:41:00" } </pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.7.GET individual payments

Description	Request to get payment status			
Method	GET/payments/{batchNumber}/paymentId			
URL	https://api-test-c2b.abb-bank.az/payments/{batchNumber}/paymentId			
Request headers	Content-type		application/json	
	Charset		UTF-8	
	Authorization		Bearer {token}	
Request parameters	Field	Type	Required	Description
	batchNumber	String (path variable)	True	Generated unique ID of file
	paymentId	String (query)	True	Generated unique ID of payment
Response type	200		Success response	
	400		Bad Request	
	500		Internal Server Error	
Success response	<pre>{ "paymentId": "2289e221-b0df-4985-8c4e-8c6c8a31cc72", "transactionReference": "120INTF220016506", "paymentStatus": { "status": "IN_PROGRESS", "transactionReference": "120INTF220016506" }, "paymentAmount": 22, "recipientAccount": "/AZ66IBAZ40140018409327445353", "paymentTime": "2022-11-30 10:40:05" }</pre>			

Error response	<pre>{ "statusCode": "ERR5", "statusDesc": "Payment with this payment number not found", "timeStamp": "2022-11-30 10:42:01" }</pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.8.GET Balance

Description	Request to get Account Balance			
Method	GET/payments/account/balance			
URL	https://api-test-c2b.abb-bank.az/payments/account/balance			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request parameters	Field	Type	Required	Description
	accountNumber	String	True	IBAN or Account number
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "accountNumber": "40050019441853946204", "currency": "AZN", "availableBalance": 0 }</pre>			
Error response	<pre>{ "statusCode": "ERR1", "statusDesc": "Unknown key was supplied. Check your input", "timeStamp": "2022-11-16 09:29:00" }</pre>			
	Field	Type	Required	Description

Error response description	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.9.GET Account Statement

Description	Request to get Account Statement			
Method	GET/payments/account/statement			
URL	https://api-test-c2b.abb-bank.az/payments/account/statement			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request parameters	Field	Type	Required	Description
	account	string	True	IBAN or Account number
	from-date	String (YYYY-MM-DD)	True	Date filter for payments (start date)
	to-date	String (YYYY-MM-DD)	True	Date filter for payments (end date)
	page-size	integer	True	Transactions in per page
	page	integer	True	page
	operation-type	string	False	<ul style="list-style-type: none"> • A • D • C <p>If Oper-type is null then by default it will be: A</p>

Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre> { "accountInfo": { "statementDate": "2023-02-23", "accountBranch": { "branchNumber": "120", "branchName": "MUSHTERI XIDMETI DEPARTAMENTI" }, "customerName": "AZERCELL TELEKOM MAHDUD MASULIYYATLI CAMIYYATI", "voen": "9900022721", "accountNumber": "40050019449310375120", "currency": "AZN" }, "transaction": { "drSum": 128717.920, "crSum": 0, "openingBalance": 36696.35, "closingBalance": 41591.43, "pageCount": 20, "currentPage": 1, "pageSize": 5, "itemsCount": 97, "transactions": [{ "trnRef": "120FTRQ211410511", "drAmount": 36696.35, "crAmount": 0, "drCr": "D", "trnDesc": "HESABDAKI QALIQ HESAB AID EDILIR", "trnDate": "21.05.2021", "counterParty": null, "beneficiaryTin": "" }, { "trnRef": "120FTRQ211440024", "drAmount": 34030.14, "crAmount": 0, "drCr": "D", "trnDesc": "HESABDAKI QALIQ HESAB AID EDILIR", </pre>			

	<pre> "trnDate": "24.05.2021", "counterParty": null, "beneficiaryTin": "" }, { "trnRef": "120FTRQ211450019", "drAmount": 57837.2, "crAmount": 0, "drCr": "D", "trnDesc": "HESABDAKI QALIG HESAB AID EDILIR", "trnDate": "25.05.2021", "counterParty": null, "beneficiaryTin": "" }, { "trnRef": "127se05211450003", "drAmount": 76.61, "crAmount": 0, "drCr": "D", "trnDesc": "FT21126Z00Q6 113IPXO211270813 AZ66IBAZ40050019449310375120 AZERCELL TELECOM MMC OB1705S/100-16 REFERENSLI MT199 ESASEN", "trnDate": "25.05.2021", "counterParty": null, "beneficiaryTin": "" }] } } </pre>			
Error response	<pre> { "statusCode": "ERR3", "statusDesc": "Debit account is not correct", "timeStamp": "2023-02-23 10:55:39" } </pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.10. POST signed Salary payments

Description	Request to send "signed" payments			
Method	POST/payments/salary/signed			
URL	https://api-test-c2b.abb-bank.az/payments/salary/signed			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre>{ "base64aDoc": "string", "externalReference": "string" }</pre>			
Request parameters	Field	Type	Required	Description
	base64aDoc	String	True	Signed adoc file
	externalReference	String	False	Unique ID
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "data": { "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57" } }</pre>			
Error response	<pre>{ "statusCode": "ERR2", "statusDesc": "File is not in base64 format", "timeStamp": "2022-11-30 10:39:16" }</pre>			

	}			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.11. POST Salary payments

Description	Request to send payments			
Method	POST/payments/salary			
URL	https://api-test-c2b.abb-bank.az/payments/salary			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request example	<pre>{ "base64aDoc": "string", "externalReference": "string" }</pre>			
Request parameters	Field	Type	Required	Description
	base64aDoc	String	True	File in "BASE64" format
	externalReference	String	False	Unique ID
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "data": { "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57" } }</pre>			
Error response	<pre>{ "statusCode": "ERR2", "statusDesc": "File is not in base64 format", "timeStamp": "2022-11-30 10:39:16" }</pre>			

	}			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

4.12. GET all salary payments

Description	Request to get payment and file status			
Method	GET/payments/salary/{batchNumber}			
URL	https://api-test-c2b.abb-bank.az/payments/salary/{batchNumber}			
Request headers	Content-type	application/json		
	Charset	UTF-8		
	Authorization	Bearer {token}		
Request parameters	Field	Type	Required	Description
	batchNumber	String	True	Generated unique ID of file
Response type	200	Success response		
	400	Bad Request		
	500	Internal Server Error		
Success response	<pre>{ "batchNumber": "284924c6-e4ba-49ce-a2b8-986b165d4c57", "status": { "status": "IN_PROGRESS", "description": "IN_PROGRESS" }, "payments": [{ "paymentId": "2289e221-b0df-4985-8c4e-8c6c8a31cc72", "transactionReference": "120INTF220016506", "paymentStatus": { "status": "IN_PROGRESS", "transactionReference": "120INTF220016506" }, "paymentAmount": 22, "recipientAccount": "/AZ66IBAZ40140018409327445353", "paymentTime": "2022-11-30 10:40:05" }] }</pre>			

	<pre> }] } </pre>			
Error response	<pre> { "statusCode": "ERR5", "statusDesc": "File with this batch number not found", "timeStamp": "2022-11-30 10:41:00" } </pre>			
Error response description	Field	Type	Required	Description
	statusCode	String	true	Status code
	statusDesc	String	true	Error message
	timestamp	String	true	Timestamp

5. XML Structure

5.1.Payment Order XML Structure

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<root>
  <payments>
    <payment>
      <!--Optional-->
      <type></type>
      <rrn></rrn>
      <!--Optional-->
      <date></date>
      <IBAN format--28 symbol>
      <account></account>
      <amount></amount>
      <!--Max 35 symbol-->
      <recipientName></recipientName>
      <!--Optional and Max 35 symbol-->
      <recipientAddress1></recipientAddress1>
      <!--Optional and Max 35 symbol-->
      <recipientAddress2></recipientAddress2>
      <!--Optional and Max 35 symbol-->
      <recipientAddress3></recipientAddress3>
      <!--Max 35 symbol-->
      <recipientAccount></recipientAccount>
      <taxId></taxId>
      <!--Optional:-->
      <pinId></pinId>
      <recipientBankCode></recipientBankCode>
      <!--Optional-->
      <feeAccount></feeAccount>
      <recipientSWIFT></recipientSWIFT>
      <!--Max 35 symbol-->
      <description1></description1>
```

```
<!--Optional and Max 35 symbol-->
<description2></description2>
<!--Optional and Max 35 symbol-->
<description3></description3>
<!--Optional and Max 35 symbol-->
<description4></description4>
<!--Optional-->
<!--Y/N-->
<urgent></urgent>
</payment>
</payments>
</root>
```

5.2. Explanation of XML structure for Payment Orders

type	Payments can be categorized as follows: Internal (IN), Domestic (DS), International Payment (FP)
rrn	Unique ID generated on your side
date	Date of payment
account	The account from which the payment is made must be specified. Debit account (in IBAN format, 28 characters)
amount	Amount of the payment
recipientName	Name of the organization receiving the funds (35 characters)
recipientAddress1	Address of the organization receiving the funds (used for international payments)
recipientAddress2	Address of the organization receiving the funds (used for international payments)
recipientAddress3	Address of the organization receiving the funds (used for international payments)
recipientAccount	Account of the organization receiving the funds (in IBAN format, 28 characters)
taxId	Tax identification number (VOEN) of the organization receiving the funds
pinId	If the recipient is a physical person, then their FIN code is written (not a mandatory field)
recipientBankCode	6-digit code of the recipient organization's bank
feeAccount	Account for deducting the commission for the payment (if no account is specified, then the account specified in the account tag will be taken by default)
recipientSWIFT	This tag is used during international payments. The bank's swift code must be specified. This tag is not used in other types of payments
description1	Payment description 1 (35 characters)
Description2	Payment description 2 (35 characters)
Description3	Payment description 3 (35 characters)
Description4	Payment description 4 (35 characters)
urgent	This tag is used during international payments. Y/N If Y, then the foreign payment is executed urgently. This tag is not used in other types of payments.

5.3.Salary Payment XML Structure

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<root>
<payments>
<payment>
<!--Mandatory-->
<rrn></rrn>
<!--Mandatory-->
<account></account>
<!--Mandatory-->
<amount></amount>
<!--Mandatory-->
<recipientAccount></recipientAccount>
<!--Mandatory and Max 35 symbol-->
<description1></description1>
<!--Optional and Max 35 symbol-->
<description2></description2>
<!--Optional and Max 35 symbol-->
<description3></description3>
<!--Optional and Max 35 symbol-->
<description4></description4>
</payment>
</payments>
</root>
```