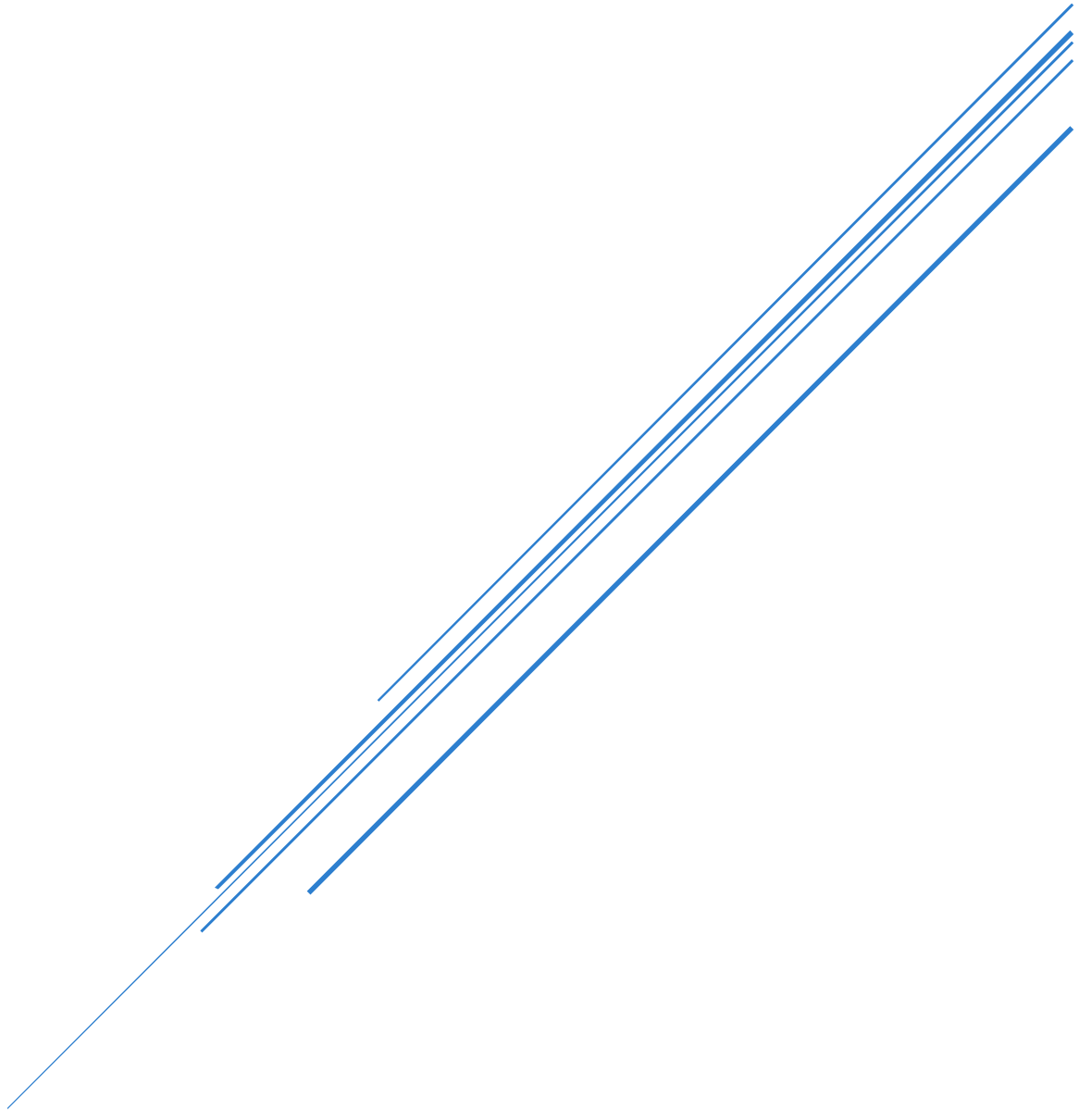


IMT(International Money Transfer)

Version 0.0.11, 16th October, 2024



Amendments

Distribution:

Version	Date Issued	Description
0.0.1	12 th October	Initial draft

Contents

Amendments.....	1
Integration: Narratives	4
Methods.....	5
1. Get Balance.....	5
Request	5
Response	5
2. Balance Movement	6
Request	6
Response	7
3. Transfer Reports	7
Request	7
Response	7
4. Transfer Report by report id	8
Request	9
Response	9
5. Get Money transfer details Files By report id	10
Request	10
Response	11
6. Get Money transfer details Files By report id and file	11
Request	11
Response	12
7. Get Credit party Information Verification For payer by payer id and Transaction Type	13
Request	13
Response	14
8. Get Available Service list.....	14
Request	14
Response	15
9. Get Available Payer list	15
Request	15

	Response	15
10.	Get Specific Payer info details by payer id	41
	Request	41
	Response	42
11.	Get Payer rate for a valid payer by payer id	47
	Request	47
	Response	48
12.	Get Countries list	48
	Request	49
	Response	49
13.	Get bank iod by swift bic code	50
	Request	50
	Response	50
14.	Get Credit party Information For payer by payer id and Transaction Type	51
	Request	51
	Response	51
15.	Create quotation.....	52
	Request	52
16.	Get Quotation by Quotation Id.....	55
	Request	55
	Response	55
17.	Get Quotation by Invoice Id	57
	Request	57
	Response	57
18.	Create Transaction By Quotation Id	59
	Request	60
19.	Create Transaction By Invoice Id	65
	Request	67
20.	Get Transaction by transaction Id	72
	Request	72
	Response	72
21.	Get Transaction by invoice Id	77

Request	77
Response	77
22. Get Confirm Transaction by invoice Id	82
Request	82
Response	82
23. Get Confirm Transaction by transaction Id	85
Request	85
Response	85
24. Send Money	88
Request	88
Response	88
HTTP Codes	91
Application/Status Codes:	92
[10] Requests	92
[20] Identification	93
[30] Sale	94
[40] Refund	94
[90] General	95

Integration: Narratives

The purpose of this document is to provide a high level specification for the IMT integration's API (Application Programming Interface). We expect the readers to have an understanding of basic API/Web Services.

. The response will be in JSON format.

The end users are expected to make their requests with valid registered merchant's credentials, to

get all the data in their scopes based on their accessibility in the system.

There will be only one web service URL be exposed and that particular web service will have various methods for clients to decide and call based on the data they require.

The web methods described below are some of the possible scenarios for the API calls and it can be extended to create as many web methods as required to service the customer needs.

APICalls:

Methods

1. Get Balance

This method or endpoint will let you know our current balance.

ContentType

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/balances

Response

Status	Response
200	Successful [{ "id": 1862, "currency": "USD", "balance": 99999337.31, "pending": 0, "available": 99999337.31, "credit_facility": 0, }]

	<pre>"name": "1862-USD" }]</pre>
--	-----------------------------------

2. Balance Movement

This method or endpoints will retrieve the amounts moved within a certain period of time. Its must not be more than 24 hours.

ContentType

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/balances/{balance_id}/movements?from_date=2024-12-01T00:00:00Z&to_date=2024-12-02T00:00:00Z

INPUT Parameters for the request JSON

Parameters	Values	Condition	Explanation	Example
balance_id	ulong	Mandatory	We can get it from balance api “id”	1862
from_date	DateTime	Mandatory	yyyy-MM-ddTHH:mm:ssZ	2024-12-01T00:00:00Z
to_date	DateTime	Mandatory	yyyy-MM-ddTHH:mm:ssZ	2024-12-02T00:00:00Z

Response

Status	Response
200	Successful

3.Transfer Reports

This method has to be called before initiating any Sale from Merchant Terminal. It returns qr_reference_code which should be used for the payment later.

ContentType

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/ccpayment/v2/money-transfer/reports?page=1&per_page=50

INPUT Parameters for the request JSON

Parameters	Values	Condition	Explanation	Example
page	integer	optional	Used for pagination and its optional	page=1
per_page	integer	optional	Used for pagination how many elements we want to see under a page	per_page=50

Response

Status	Response
--------	----------

200

Successful

```
[
  {
    "id": 1456304,
    "type": "PARTNER_TRANSACTION_REPORT_DAILY",
    "timezone": "UTC",
    "cut_off_time": "00:00:00",
    "report_date": "2024-08-11",
    "period_start": "2024-08-11T00:00:00Z",
    "period_end": "2024-08-12T00:00:00Z",
    "files": [
      {
        "id": 2625450,
        "report_type": "PARTNER_TRANSACTION_REPORT_DAILY",
        "file_type": "TRANSACTION_STATS",
        "name": "daily_transaction_report_2024-08-11.csv"
      }
    ]
  },
  {
    "id": 1447007,
    "type": "PARTNER_TRANSACTION_REPORT_DAILY",
    "timezone": "UTC",
    "cut_off_time": "00:00:00",
    "report_date": "2024-08-06",
    "period_start": "2024-08-06T00:00:00Z",
    "period_end": "2024-08-07T00:00:00Z",
    "files": [
      {
        "id": 2609651,
        "report_type": "PARTNER_TRANSACTION_REPORT_DAILY",
        "file_type": "TRANSACTION_STATS",
        "name": "daily_transaction_report_2024-08-06.csv"
      }
    ]
  }
]
```

4. Transfer Report by report id

By calling this api we will get the transfer report by the report id.

ContentType

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/reports/{id}

INPUT Parameters for the request JSON

Parameters	Schema	Condition	Explanation	Examples
id	integer	Mandatory	This should be the report id	1450478

Response

200	<div>Success</div> <div>{</div> <div> "id": 1458311,</div> <div> "type": "PARTNER_ACCOUNT_STATEMENT_DAILY",</div> <div> "timezone": "UTC",</div> <div> "cut_off_time": "00:00:00",</div> <div> "report_date": "2024-08-11",</div> <div> "period_start": "2024-08-11T00:00:00Z",</div> <div> "period_end": "2024-08-12T00:00:00Z",</div> <div> "files": [</div> <div> {</div> <div> "id": 2630002,</div> <div> "report_type": "PARTNER_ACCOUNT_STATEMENT_DAILY",</div> <div> "file_type": "AGGREGATED_STATS",</div> <div> "name": "daily_account_statement_aggregation_2024-08-11.csv"</div> <div> },</div> <div> {</div> <div> "id": 2630003,</div> <div> "report_type": "PARTNER_ACCOUNT_STATEMENT_DAILY",</div> <div> "file_type": "UNDERLYING_STATS",</div>
-----	--

	<pre> "name": "daily_account_statement_underlying_1862_USD_2024-08-11.csv" }] } </pre>
400	<p>Failure</p> <pre> [{ "code": "1009001", "message": "Unexpected error, please contact our support team" }] </pre>

5. Get Money transfer details Files By report id

By Calling this end point we will get the money transfer details files by the report id.

Content Type

application/json

Request

Method	URL
GET	<p>Test</p> <p>https://imt.sipay.com.tr/v2/money-transfer/reports/{id}/files</p>

INPUT Parameters for the request JSON

Parameters	Schema	Condition	Explanation	Examples
id	integer	Mandatory	This should be the report id	1458311

Response

200	Success [{ "id": 2615717, "report_type": "PARTNER_TRANSACTION_REPORT_DAILY", "file_type": "TRANSACTION_STATS", "name": "daily_transaction_report_2024-08-08.csv" }]
200	Failure []

6. Get Money transfer details Files By report id and file

By Calling this end point we will get the money transfer details fils by the report id along with the file specific id

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/reports/{report_id}/files/{id}

INPUT Parameters for the request JSON

Parameters	Schema	Condition	Explanation	Examples
report_id	integer	Mandatory	This should be the report id	1458311
id	integer	Mandatory	This should be the specific file id for that specific report	

Response

200

Success

{

"url": "https://cobra-reports.s3.eu-west-1.amazonaws.com/preproduction/external/partner_daily_account_statements/1279/2024-08-11/daily_account_statement_aggregation_2024-08-11.csv?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASIAYO4ZOUJOU6T4F6U%2F20240815%2Feu-west-1%2Fs3%2Faws4_request&X-Amz-Date=20240815T095451Z&X-Amz-Expires=600&X-Amz-Security-Token=IQoJb3JpZ2luX2VjENr%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2FwEaCWV1LXdlc3QtMSJGMEQCIDQbRVANLb0465SFmCYkb9HYfa%2FXrfyYsK4FCQASngVUAiAcK0W0ux8X1hgRElayQpqmOHBUfrlssKu%2BfjuCIQbAiir5BAJT%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F8BEAMaDDU4MTc1MzAyMTMzMCImaf68u8Z7%2BKKQgFiH4Ks0Eg4vLOqDhzA%2BCWxadlZ%2FBPGdZEMxl4nM4x2Y5CbTuwKqkBUJ8x%2Bn9OkcR%2FQ1%2BxEfsP%2B1b02dB59cQRXx6zlG7Jff10XJXblN5HFq3d0aoQKz0%2Fq8Bci8W2A%2FyQYS4I6YU3ipd3Oo3pqAoFu%2B9IR0%2BCY3ZbnPhdg9v%2Fy%2Fa2KpojjymRfuZ1Nj%2F5559uFM8COFFXuRj0mj2Yc5urweQ06oOFgNSFA53cGlvWzgOmKjdjxbqndlB8SLGZ60MyKKZvVIG5wTjwqYT9xjxmzZH06xrc5MJQDFR%2B%2BR0LKtkWyAXu7EZXFWzMCbYgtD6%2FTckYJrKqj1n6Y3upXcbRsN1jiXBdihYBx%2BmZMBgn4SspOEmQGHlXz1im0y81v6E0R%2FPKeshSHWH2G4JUkw2Zya9AB0GKG1Ez9GqkLQ%2FuDJfvAYoKlX6yQYeyzDzHaUkhg00LVU04JMeE2egh9dl0GBIB6cWBZo8WjGoYY%2BVVqcr5t63Yj9%2F21DX92Vwle7A8ld6gjZPtseCGoMRYPwB2gHJz4guu%2B2MWm7zsilslDx7pwVEkSEN%2Fm2V1RK4pl6N%2BERXLz1EgK0e0kq83%2B9suRtZPZZMn6anWS5KNUKVVfVxCDijiOOpQ3vTBQTugULYJbpusKiwA7nd%2FYoluxOy4eqsdN3lpeByQwgrDZOLsctUgxR8M4QE5XEMfgXSzld2uJY7aPYOLOvNizss8turseXfkP6LQFm10RoBAbyAevMYkzfM2Ok0wOLejT0TFhz%2BwpmNCnSxolrPZSacjawVF%2Fn0OzDroPe1BjqbAdBvMpyYoFK16nOMOOe68vk3Q%2By7TBWZGvksUIYs20SDElwgY5UjYkuK5WoLSa8yq7VmMgsa1mipmCfie%2BKXchDF0ZTsQdtSbt03p1ZRcyqepCovDj4PaykRtbrY37%2FkPLY%2FNvd8n18%2Bcue3G0tCL6DiZhr5LL6KvkGel8Nsh1ng5g6wB3GVaH8ImYE%2FJMWTqMDOgZqasE5eMJMe&X-Amz-SignedHeaders=host&response-content-disposition=inline%3B%20filename%3D%22preproduction%2Fexternal%2Fpartner_daily_account_statements%2F1279%2F2024-08-11%2Fdaily_account_statement_aggregation_2024-08-

	11.csv%22&X-Amz-Signature=764b9fa00be44fabe718079d1135ca52c6f4a2e5d40acc afe68d049d24c103a4" }
400	[{ "code": "1009001", "message": "Unexpected error, please contact our support team" }]

7. Get Credit party Information Verification For payer by payer id and Transaction Type

By Calling this end point we can verify the credit party information by the payer id and the transaction type passed on query parameter and we have to pass the msisdn in the request body we don't have any msisdn contract yet with thunes but we implemented this end point as well.

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/payers/{id}/{transaction_type}/credit-party-verification

Parameters	Schema	Condition	Explanation	Examples
transaction_type	string	Mandatory	This should be a string "B2B", "B2C", "C2C", "B2B"	B2B
id	integer	Mandatory	This should be the payer id	1807

Request Body:

```
{
  "credit_party_identifier": {
    "msisdn": ""
  }
}
```

Response

400	[{ "code": "1007100", "message": "Method not supported for this payer" }]
-----	--

8. Get Available Service list

By Calling this end point we can get the service list as well for us and opur contract its only bank account till now.

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/services

Response

200	[{ "id": 2, "name": "BankAccount" }]
-----	---

9. Get Available Payer list

By Calling this end point we can get the payer list.

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/payers

Response

200	[{ "id": 1807, "name": "All Banks / GBR / UK Faster Payment", }]
-----	---


```
"precision": 2,

"increment": 0.01,

"currency": "GBP",

"country_iso_code": "GBR",

"service": {

  "id": 2,

  "name": "BankAccount"

},

"transaction_types": {

  "b2B": {

    "credit_party_identifiers_accepted": [

      [

        "sort_code",

        "account_number"

      ]

    ],

    "credit_party_information": {

      "credit_party_identifiers_accepted": [

        []

      ]

    },

    "credit_party_verification": {

      "credit_party_identifiers_accepted": [

        []

      ],

      "required_receiving_entity_fields": [

        []

      ]

    }

  }

}
```

```
},
"maximum_transaction_amount": null,
"minimum_transaction_amount": "0.00000000",
"purpose_of_remittance_values_accepted": [],
"required_documents": [
[]
],
"required_receiving_entity_fields": [
[
"registered_name",
"country_iso_code"
]
],
"required_sending_entity_fields": [
[
"registered_name",
"country_iso_code",
"code",
"registration_number",
"address",
"city"
]
]
},
"b2C": {
"credit_party_identifiers_accepted": [
[
"sort_code",
```

	<pre>"account_number"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"]]</pre>
--	--

```
],
"required_sending_entity_fields": [
  [
    "registered_name",
    "country_iso_code",
    "code",
    "registration_number"
  ]
],
},
"c2C": {
  "credit_party_identifiers_accepted": [
    [
      "sort_code",
      "account_number"
    ],
    [
      "iban",
      "swift_bic_code"
    ],
    [
      "iban"
    ]
  ],
  "credit_party_information": {
    "credit_party_identifiers_accepted": [
      []
    ]
  }
}
```

	<pre>}, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]], }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"]], "required_sending_entity_fields": [["lastname", "firstname", "code", "date_of_birth",</pre>
--	--

	<pre>"country_of_birth_iso_code", "address", "city", "country_iso_code"]] } } }, { "id": 1809, "name": "All Banks Germany / EUR / Payment System: SEPA", "precision": 2, "increment": 0.01, "currency": "EUR", "country_iso_code": "DEU", "service": { "id": 2, "name": "BankAccount" }, "transaction_types": { "b2B": { "credit_party_identifiers_accepted": [["iban"]], "credit_party_information": {</pre>
--	---

	<pre>"credit_party_identifiers_accepted": [[]], "credit_party_verification": { "credit_party_identifiers_accepted": [[], "required_receiving_entity_fields": [[] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[], "required_receiving_entity_fields": [["registered_name", "country_iso_code"], ["registered_name", "address", "city", "country_iso_code"</pre>
--	--

```
]
],
"required_sending_entity_fields": [
[
"registered_name",
"country_iso_code",
"code",
"registration_number"
],
[
"registered_name",
"country_iso_code",
"code",
"registration_number",
"address",
"city"
]
]
},
"b2C": {
"credit_party_identifiers_accepted": [
[
"iban"
]
],
"credit_party_information": {
"credit_party_identifiers_accepted": [
[]
]
```


	<pre>] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]]</pre>
--	--

```
]
],
"required_sending_entity_fields": [
[
"registered_name",
"country_iso_code",
"code",
"registration_number"
],
[
"registered_name",
"country_iso_code",
"code",
"registration_number",
"address",
"city"
]
]
},
"c2C": {
"credit_party_identifiers_accepted": [
[
"iban"
]
],
"credit_party_information": {
"credit_party_identifiers_accepted": [
[]
]
```

	<pre>] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]]</pre>
--	--

```
]
],
"required_sending_entity_fields": [
[
"lastname",
"firstname",
"code",
"date_of_birth",
"country_of_birth_iso_code",
"country_iso_code"
],
[
"lastname",
"firstname",
"code",
"date_of_birth",
"country_of_birth_iso_code",
"address",
"city",
"country_iso_code"
]
]
}
},
{
"id": 1995,
"name": "All Banks Netherlands / EUR / Payment System: SEPA",
```

```
"precision": 2,

"increment": 0.01,

"currency": "EUR",

"country_iso_code": "NLD",

"service": {

  "id": 2,

  "name": "BankAccount"

},

"transaction_types": {

  "b2B": {

    "credit_party_identifiers_accepted": [

      [

        "iban"

      ]

    ],

    "credit_party_information": {

      "credit_party_identifiers_accepted": [

        []

      ]

    },

    "credit_party_verification": {

      "credit_party_identifiers_accepted": [

        []

      ],

      "required_receiving_entity_fields": [

        []

      ]

    },

  },

}
```

	<pre>"maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["registered_name", "country_iso_code"], ["registered_name", "address", "city", "country_iso_code"]], "required_sending_entity_fields": [["registered_name", "country_iso_code", "code", "registration_number"], ["registered_name", "country_iso_code",</pre>
--	--

	<pre>"code", "registration_number", "address", "city"]], }, "b2C": { "credit_party_identifiers_accepted": [["iban"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000",</pre>
--	--

	<pre>"purpose_of_remittance_values_accepted": [], "required_documents": [[],], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]], "required_sending_entity_fields": [["registered_name", "country_iso_code", "code", "registration_number"], ["registered_name", "country_iso_code",</pre>
--	---

	<pre>"code", "registration_number", "address", "city"]], }, "c2C": { "credit_party_identifiers_accepted": [["iban"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000",</pre>
--	--

	<pre>"purpose_of_remittance_values_accepted": [], "required_documents": [[],], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]], "required_sending_entity_fields": [["lastname", "firstname", "code", "date_of_birth", "country_of_birth_iso_code", "country_iso_code"], []</pre>
--	---

	<pre>"lastname", "firstname", "code", "date_of_birth", "country_of_birth_iso_code", "address", "city", "country_iso_code"]] } }, { "id": 5578, "name": "All Banks United Kingdom / EUR / Payment System: SEPA", "precision": 2, "increment": 0.01, "currency": "EUR", "country_iso_code": "GBR", "service": { "id": 2, "name": "BankAccount" }, "transaction_types": { "b2B": { "credit_party_identifiers_accepted": [</pre>
--	--

	<pre>["iban"], ["iban", "swift_bic_code"] }, "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [</pre>
--	--

```
[  
  "registered_name",  
  "country_iso_code"  
],  
[  
  "registered_name",  
  "address",  
  "city",  
  "country_iso_code"  
]  
],  
"required_sending_entity_fields": [  
  [  
    "registered_name",  
    "country_iso_code",  
    "code",  
    "registration_number"  
  ],  
  [  
    "registered_name",  
    "country_iso_code",  
    "code",  
    "registration_number",  
    "address",  
    "city"  
  ]  
]  
},
```

```
"b2C": {  
  "credit_party_identifiers_accepted": [  
    [  
      "iban"  
    ],  
    [  
      "iban",  
      "swift_bic_code"  
    ]  
  ],  
  "credit_party_information": {  
    "credit_party_identifiers_accepted": [  
      []  
    ],  
  },  
  "credit_party_verification": {  
    "credit_party_identifiers_accepted": [  
      []  
    ],  
    "required_receiving_entity_fields": [  
      []  
    ],  
  },  
  "maximum_transaction_amount": null,  
  "minimum_transaction_amount": "0.00000000",  
  "purpose_of_remittance_values_accepted": [],  
  "required_documents": [  
    []  
  ]  
}
```

	<pre>], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]], "required_sending_entity_fields": [["registered_name", "country_iso_code", "code", "registration_number"], ["registered_name", "country_iso_code", "code", "registration_number", "address",</pre>
--	---

	<pre> "city"]] }, "c2C": { "credit_party_identifiers_accepted": [["iban"], ["iban", "swift_bic_code"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, </pre>
--	---

	<pre>"minimum_transaction_amount": "0.00000000", "purpose_of_remittance_values_accepted": [], "required_documents": [[]], "required_receiving_entity_fields": [["lastname", "firstname", "country_iso_code"], ["lastname", "firstname", "address", "city", "country_iso_code"]], "required_sending_entity_fields": [["lastname", "firstname", "code", "date_of_birth", "country_of_birth_iso_code", "country_iso_code"]],</pre>
--	---

	<pre>["lastname", "firstname", "code", "date_of_birth", "country_of_birth_iso_code", "address", "city", "country_iso_code"]] } } }</pre>
--	---

10. Get Specific Payer info details by payer id

By Calling this end point we can get the payer details info by payer id.

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/payers/{id}

INPUT Parameters for the request JSON

Parameters	Schema	Condition	Explanation	Examples
id	integer	Mandatory	This should be the payer id	1807

Response

200	<pre>{ "id": 1807, "name": "All Banks / GBR / UK Faster Payment", "precision": 2, "increment": 0.01, "currency": "GBP", "country_iso_code": "GBR", "service": { "id": 2, "name": "BankAccount" }, "transaction_types": { "b2B": { "credit_party_identifiers_accepted": [["sort_code", "account_number"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] } } } }</pre>
-----	---

```
},
"credit_party_verification": {
  "credit_party_identifiers_accepted": [
    []
  ],
  "required_receiving_entity_fields": [
    []
  ]
},
"maximum_transaction_amount": null,
"minimum_transaction_amount": "0",
"purpose_of_remittance_values_accepted": [],
"required_documents": [
  []
],
"required_receiving_entity_fields": [
  [
    "registered_name",
    "country_iso_code"
  ]
],
"required_sending_entity_fields": [
  [
    "registered_name",
    "country_iso_code",
    "code",
    "registration_number",
    "address",
```

	<pre>"city"]] }, "b2C": { "credit_party_identifiers_accepted": [["sort_code", "account_number"]], "credit_party_information": { "credit_party_identifiers_accepted": [[]] }, "credit_party_verification": { "credit_party_identifiers_accepted": [[]], "required_receiving_entity_fields": [[]] }, "maximum_transaction_amount": null, "minimum_transaction_amount": "0", "purpose_of_remittance_values_accepted": [], "required_documents": [</pre>
--	--

```
[  
  ],  
  "required_receiving_entity_fields": [  
    [  
      "lastname",  
      "firstname",  
      "country_iso_code"  
    ]  
  ],  
  "required_sending_entity_fields": [  
    [  
      "registered_name",  
      "country_iso_code",  
      "code",  
      "registration_number"  
    ]  
  ]  
},  
"c2C": {  
  "credit_party_identifiers_accepted": [  
    [  
      "sort_code",  
      "account_number"  
    ],  
    [  
      "iban",  
      "swift_bic_code"  
    ],  
  ],  
}
```

```
[
  "iban"
],
"credit_party_information": {
  "credit_party_identifiers_accepted": [
    []
  ],
  "credit_party_verification": {
    "credit_party_identifiers_accepted": [
      []
    ],
    "required_receiving_entity_fields": [
      []
    ],
    "maximum_transaction_amount": null,
    "minimum_transaction_amount": "0",
    "purpose_of_remittance_values_accepted": [],
    "required_documents": [
      []
    ],
    "required_receiving_entity_fields": [
      [
        "lastname",
        "firstname",
        "country_iso_code"
      ]
    ]
  }
}
```

	<pre>]], "required_sending_entity_fields": [["lastname", "firstname", "code", "date_of_birth", "country_of_birth_iso_code", "address", "city", "country_iso_code"]] } } } </pre>
--	--

11. Get Payer rate for a valid payer by payer id

By Calling this end point we can get the payer rate for a specific payer by payer id

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/payers/{id}/rates

Parameters	Schema	Condition	Explanation	Examples
id	integer	Mandatory	This should be the payer id	1807

Response

200	Success <pre>{ "destination_currency": "GBP", "rates": { "c2C": { "eur": null } } }</pre>
400	Failure <pre>[{ "code": "1003002", "message": "Invalid payer" }]</pre>

12. Get Countries list

By Calling this end point we can get the list of the available countries.

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/countries

Response

200	Success [{ "iso_code": "DEU", "name": "Germany" }, { "iso_code": "GBR", "name": "United Kingdom of Great Britain and Northern Ireland" }, { "iso_code": "NLD", "name": "Netherlands" }]
-----	--

13. Get bank id by swift bic code

By Calling this end point we can get bank id by the swift bic code.

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/lookups/BIC/{swift_bic_code}

Parameters	Values	Condition	Explanation	Example
swift_bic_code	string	Mandatory	We have the pass a valid bank bank swift bic copde	ABNAIDJA

Response

200	Success [{ "id": 123 }]
200	Failure []

14. Get Credit party Information For payer by payer id and Transaction Type

By Calling this end point we can get the credit party information by the payer id and the transaction type passed on query parameter and we have to pass the msisdn in the request body we don't have any msisdn contract yet with thunes but we implemented this end point as well.

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/payers/{id}/{transaction_type}/credit-party-information

Parameters	Schema	Condition	Explanation	Examples
transaction_type	string	Mandatory	This should be a string "B2B","B2C","C2C","B2B"	B2B
id	integer	Mandatory	This should be the payer id	1807

Request Body:

```
{
  "credit_party_identifier": {
    "msisdn": ""
  }
}
```

Response

400	[
-----	---

	<pre>{ "code": "1007100", "message": "Method not supported for this payer" }</pre>
--	--

15. Create quotation

By calling this end point we will be able to create a quotation it could be three types B2B,B2C,C2C And the mode can be two types SOURCE_AMOUNT,DESTINATION_AMOUINT

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/quotations

Request Body:

```
{
  "invoice_id": "B2B5339", //unique id should be pass here
  "payer_id": "1809", // we will get the payer id from payer endpoint
  "customer_id": "1809", // if its white label customer then the customer id should be pass here
  "mode": "SOURCE_AMOUNT", //could be two type source_amount,destination_amount
  "transaction_type": "B2B", //could be three type b2b,b2c,c2c
  "source_amount": 1, //if mode source amount then this must be passed
  "source_currency_code": "USD", // source currency should be USD always
}
```

```
"source_country_iso_code": "TUR", //source country iso code should be TUR always
"destination_amount": 1, //if mode destination amount then destination amount must be passed
"destination_currency_code": "EUR" //destination currency must be passed if mode destination
//amount
}
```

Response

200	<pre>{ "id": 61014900, "external_id": "B2B5339", "invoice_id": " B2B5339", "payer": { "id": 1809, "name": "All Banks Germany / EUR / Payment System: SEPA", "currency": "EUR", "country_iso_code": "DEU", "service": { "id": 2, "name": "BankAccount" } }, "mode": "SOURCE_AMOUNT", "transaction_type": "B2B", "source": { "country_iso_code": "TUR", "currency": "USD", "amount": 1 }, "destination": { "currency": "EUR",</pre>
-----	---

	<pre>"amount": 0.9 }, "sent_amount": { "currency": "USD", "amount": 1 }, "wholesale_fx_rate": 0.903579845472637, "fee": { "currency": "USD", "amount": 1.5 }, "creation_date": "2024-08-15T12:15:27Z", "expiration_date": "2024-08-15T13:15:27Z" }</pre>
400	<pre>[{ "code": "1007001", "message": "External ID already used" }]</pre>

16. Get Quotation by Quotation Id

*By Calling this end point we can get the quotation details by the quotation id.
In the query parameter we have to pass the valid quotation id.*

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr /v2/money-transfer/quotations/{id}

Response

200	Success { "id": 61014794, "external_id": "B2B53362", "invoice_id": " B2B53362", "payer": { "id": 1809, "name": "All Banks Germany / EUR / Payment System: SEPA", "currency": "EUR", "country_iso_code": "DEU", "service": { "id": 2, "name": "BankAccount" } }, "mode": "DESTINATION_AMOUNT", }
-----	--

	<pre>"transaction_type": "B2C", "source": { "country_iso_code": "TUR", "currency": "USD", "amount": 1.11 }, "destination": { "currency": "EUR", "amount": 1 }, "sent_amount": { "currency": "USD", "amount": 1.11 }, "wholesale_fx_rate": 0.903579845472637, "fee": { "currency": "USD", "amount": 1.5 }, "creation_date": "2024-08-15T12:14:41Z", "expiration_date": "2024-08-15T13:14:41Z" }</pre>
--	--

17. Get Quotation by Invoice Id

*By Calling this end point we can get the quotation details by the invoice id.
In the query parameter we have to pass the valid unique invoice id.
Invoice Id must be a number.*

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/quotations/ext-{invoice_id}

Response

200	Success { "id": 61017135, "external_id": "1011", "invoice_id": "1011", "payer": { "id": 1809, "name": "All Banks Germany / EUR / Payment System: SEPA", "currency": "EUR", "country_iso_code": "DEU", "service": { "id": 2, "name": "BankAccount" } } },
-----	--

	<pre>"mode": "DESTINATION_AMOUNT", "transaction_type": "B2C", "source": { "country_iso_code": "TUR", "currency": "USD", "amount": 1.11 }, "destination": { "currency": "EUR", "amount": 1 }, "sent_amount": { "currency": "USD", "amount": 1.11 }, "wholesale_fx_rate": 0.903579845472637, "fee": { "currency": "USD", "amount": 1.5 }, "creation_date": "2024-08-15T12:29:58Z", "expiration_date": "2024-08-15T13:29:58Z" }</pre>
--	--

18. Create Transaction By Quotation Id

By calling this end point we will be able to create a transaction . We can create mainly 3 types of transaction here B2B,B2C,C2C according to transaction type and 2 types of transaction according to mode SOURCE_AMOUNT,DESTINATION_AMOUNT

I am giving the different valid request

1. B2B (Demo Request Body)

```
{
  "external_id": "1234", //valid invoice id here what input was in create quotation
  "credit_party_identifier": {
    "iban": "DE05975683053163247100" //valid credit party identifiers iban (international
    //bank account number)
  },
  "sending_business": {
    "code": "Code Placeholder", //sending business code should be a string
    "country_iso_code": "TUR", //sending business iso code should be TUR in our case
    "registered_name": "ABC COMPANY", //registered company name
    "registration_number": "ABC12345T" //registered company number
  },
  "receiving_business": {
    "registered_name": "XYZ COMPANY", //receiver business company name
    "country_iso_code": "DEU" //receiver business company country iso code
  },
  "document_reference_number": "ABC1234-9876", //put a document reference number
  "purpose_of_remittance": "ADVISORY_FEES", //put a purpose for the remittance
  "url": https://imt.sipay.com.tr/api/v1/callback //put this url
}
```

2.B2C (Demo Request Body)

```
{
  "external_id": "1235",
  "beneficiary": {
    "firstname": "Duncan",
    "country_iso_code": "NLD",
    "lastname": "Ruben"
  },
  "sending_business": {
    "code": "code placeholder...",
    "country_iso_code": "TUR",
    "registered_name": "ABC COMPANY",
```

```

    "registration_number": "ABC12345T"
  },
  "credit_party_identifier": {
    "iban": "NL79VRS8450806100"
  },
  "purpose_of_remittance": "ADVISORY_FEES",
  "url": "https://imt.sipay.com.tr/api/v1/callback"
}

```

3.C2C (Demo Request Body)

```

{
  "credit_party_identifier": {
    "iban": "NL79VRS8450806100"
  },
  "sender": {
    "address": "12B Baker Street",
    "country_iso_code": "TUR",
    "city": "Amsterdam",
    "lastname": "Smith",
    "date_of_birth": "1980-12-19",
    "code": "code placeholder...",
    "firstname": "Smith",
    "country_of_birth_iso_code": "TUR"
  },
  "beneficiary": {
    "lastname": "Anderson",
    "firstname": "Anderson",
    "country_iso_code": "TUR"
  },
  "purpose_of_remittance": "ADVISORY_FEES",
  "external_id": "1236"
}

```

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/quotations/{{quotation_id}}/transactions

Response

```
200 {  
  "id": 4367150,  
  "status": "10000",  
  "status_message": "CREATED",  
  "status_class": "1",  
  "status_class_message": "CREATED",  
  "external_id": "1234",  
  "invoice_id": "1234",  
  "external_code": null,  
  "transaction_type": "B2B",  
  "payer_transaction_reference": null,  
  "payer_transaction_code": null,  
  "creation_date": "2024-08-15T13:41:53Z",  
  "expiration_date": "2024-08-15T14:31:34Z",  
  "credit_party_identifier": {  
    "msisdn": null,  
    "bank_account_number": null,  
    "swift_bic_code": null,  
    "iban": "DE05975683053163247100"  
  },  
  "source": {  
    "country_iso_code": "TUR",  
    "currency": "USD",  
    "amount": 1  
  },  
  "destination": {
```

```
"currency": "EUR",
"amount": 0.91
},
"payer": {
  "id": 1809,
  "name": "All Banks Germany / EUR / Payment System: SEPA",
  "currency": "EUR",
  "country_iso_code": "DEU",
  "service": {
    "id": 2,
    "name": "BankAccount"
  }
},
"sender": null,
"beneficiary": null,
"callback_url": null,
"sent_amount": {
  "currency": "USD",
  "amount": 1
},
"wholesale_fx_rate": 0.907762772835821,
"retail_rate": null,
"retail_fee": 0,
"retail_fee_currency": null,
"fee": {
  "currency": "USD",
  "amount": 1.5
},
```

	<p>"purpose_of_remittance": "ADVISORY_FEES",</p> <p>"document_reference_number": "ABC1234-9876",</p> <p>"additional_information_1": null,</p> <p>"additional_information_2": null,</p> <p>"additional_information_3": null,</p> <p>"reference": null,</p> <p>"receiving_business": {</p> <p> "address": null,</p> <p> "city": null,</p> <p> "code": null,</p> <p> "country_iso_code": "DEU",</p> <p> "date_of_incorporation": null,</p> <p> "email": null,</p> <p> "msisdn": null,</p> <p> "postal_code": null,</p> <p> "province_state": null,</p> <p> "registered_name": "XYZ COMPANY",</p> <p> "registration_number": null,</p> <p> "representative_firstname": null,</p> <p> "representative_id_country_iso_code": null,</p> <p> "representative_id_delivery_date": null,</p> <p> "representative_id_expiration_date": null,</p> <p> "representative_id_number": null,</p> <p> "representative_id_type": null,</p> <p> "representative_lastname": null,</p> <p> "representative_lastname2": null,</p> <p> "representative_middlename": null,</p> <p> "representative_nativename": null,</p>
--	--

	<pre>"tax_id": null, "trading_name": null }, "sending_business": { "address": null, "business_relationship": null, "city": null, "code": "Code Placeholder", "country_iso_code": "TUR", "date_of_incorporation": null, "email": null, "msisdn": null, "postal_code": null, "province_state": null, "registered_name": "ABC COMPANY", "registration_number": "ABC12345T", "representative_firstname": null, "representative_id_country_iso_code": null, "representative_id_delivery_date": null, "representative_id_expiration_date": null, "representative_id_number": null, "representative_id_type": null, "representative_lastname": null, "representative_lastname2": null, "representative_middlename": null, "representative_nativename": null, "tax_id": null, "trading_name": null</pre>
--	---

	<pre> } } </pre>
400	<pre> [{ "code": "1007001", "message": "External ID already used" }] </pre>

19. Create Transaction By Invoice Id

By calling this end point we will be able to create a transaction . We can create mainly 3 types of transaction here B2B,B2C,C2C according to transaction type and 2 types of transaction according to mode SOURCE_AMOUNT,DESTINATION_AMOUNT

I am giving the different valid request

1. B2B (Demo Request Body)

```

{
  "invoice_id": "1234", //valid invoice id here what input was in create quotation
  "credit_party_identifier": {
    "iban": "DE05975683053163247100" //valid credit party identifiers iban (international
                                     //bank account number)
  },
  "sending_business": {
    "code": "Code Placeholder", //sending business code should be a string
    "country_iso_code": "TUR", //sending business iso code should be TUR in our case
    "registered_name": "ABC COMPANY", //registered company name
    "registration_number": "ABC12345T" //registered company number
  }
}

```

},

```
"receiving_business": {  
  "registered_name": "XYZ COMPANY", //receiver business company name  
  "country_iso_code": "DEU" //receiver business company country iso code  
},  
"document_reference_number": "ABC1234-9876", //put a document reference number  
"purpose_of_remittance": "ADVISORY_FEES", //put a purpose for the remittance  
"url": https://imt.sipay.com.tr/api/v1/callback //put this url  
}
```

2.B2C (Demo Request Body)

```
{  
  "invoice_id": "1235",  
  "beneficiary": {  
    "firstname": "Duncan",  
    "country_iso_code": "NLD",  
    "lastname": "Ruben"  
  },  
  "sending_business": {  
    "code": "code placeholder...",  
    "country_iso_code": "TUR",  
    "registered_name": "ABC COMPANY",  
    "registration_number": "ABC12345T"  
  },  
  "credit_party_identifier": {  
    "iban": "NL79VRS8450806100"  
  },  
  "purpose_of_remittance": "ADVISORY_FEES",  
  "url": "https://imt.sipay.com.tr/api/v1/callback"  
}
```

3.C2C (Demo Request Body)

```
{  
  "credit_party_identifier": {  
    "iban": "NL79VRS8450806100"  
  },  
  "sender": {  
    "address": "12B Baker Street",  
    "country_iso_code": "TUR",  
    "city": "Amsterdam",  
    "lastname": "Smith",  
    "date_of_birth": "1980-12-19",  
    "code": "code placeholder..."
```

```
"firstname": "Smith",  
"country_of_birth_iso_code": "TUR"  
},  
"beneficiary": {  
  "lastname": "Anderson",  
  "firstname": "Anderson",  
  "country_iso_code": "TUR"  
},  
"purpose_of_remittance": "ADVISORY_FEES",  
"invoice_id": "1236"  
}
```

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/transactions/ext-{external_id}

Response

200	{ "id": 4367150, "status": "10000", "status_message": "CREATED", "status_class": "1", "status_class_message": "CREATED", "external_id": "1234", "invoice_id": "1234", "external_code": null, "transaction_type": "B2B", "payer_transaction_reference": null, }
-----	---

```
"payer_transaction_code": null,

"creation_date": "2024-08-15T13:41:53Z",

"expiration_date": "2024-08-15T14:31:34Z",

"credit_party_identifier": {

  "msisdn": null,

  "bank_account_number": null,

  "swift_bic_code": null,

  "iban": "DE05975683053163247100"

},

"source": {

  "country_iso_code": "TUR",

  "currency": "USD",

  "amount": 1

},

"destination": {

  "currency": "EUR",

  "amount": 0.91

},

"payer": {

  "id": 1809,

  "name": "All Banks Germany / EUR / Payment System: SEPA",

  "currency": "EUR",

  "country_iso_code": "DEU",

  "service": {

    "id": 2,

    "name": "BankAccount"

  }

},
```

	<pre>"sender": null, "beneficiary": null, "callback_url": null, "sent_amount": { "currency": "USD", "amount": 1 }, "wholesale_fx_rate": 0.907762772835821, "retail_rate": null, "retail_fee": 0, "retail_fee_currency": null, "fee": { "currency": "USD", "amount": 1.5 }, "purpose_of_remittance": "ADVISORY_FEES", "document_reference_number": "ABC1234-9876", "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "reference": null, "receiving_business": { "address": null, "city": null, "code": null, "country_iso_code": "DEU", "date_of_incorporation": null, "email": null,</pre>
--	---

	<pre>"msisdn": null, "postal_code": null, "province_state": null, "registered_name": "XYZ COMPANY", "registration_number": null, "representative_firstname": null, "representative_id_country_iso_code": null, "representative_id_delivery_date": null, "representative_id_expiration_date": null, "representative_id_number": null, "representative_id_type": null, "representative_lastname": null, "representative_lastname2": null, "representative_middlename": null, "representative_nativename": null, "tax_id": null, "trading_name": null }, "sending_business": { "address": null, "business_relationship": null, "city": null, "code": "Code Placeholder", "country_iso_code": "TUR", "date_of_incorporation": null, "email": null, "msisdn": null, "postal_code": null,</pre>
--	--

	<pre>"province_state": null, "registered_name": "ABC COMPANY", "registration_number": "ABC12345T", "representative_firstname": null, "representative_id_country_iso_code": null, "representative_id_delivery_date": null, "representative_id_expiration_date": null, "representative_id_number": null, "representative_id_type": null, "representative_lastname": null, "representative_lastname2": null, "representative_middlename": null, "representative_nativename": null, "tax_id": null, "trading_name": null } }</pre>
400	<pre>[{ "code": "1007001", "message": "External ID already used" }]</pre>

20. Get Transaction by transaction Id

*By Calling this end point we can get the transaction details by the transaction id.
In the query parameter we have to pass the valid transaction id(e.g 4368838).*

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr/v2/money-transfer/transactions/{id}

Response

200	Success { "id": 4368838, "status": "70000", "status_message": "COMPLETED", "status_class": "7", "status_class_message": "COMPLETED", "external_id": "12001C2C", "external_code": null, "transaction_type": "C2C", "payer_transaction_reference": "1723791907", "payer_transaction_code": null, "creation_date": "2024-08-16T07:05:03Z", "expiration_date": "2024-08-16T08:05:03Z", }
-----	---

```
"credit_party_identifier": {  
  "msisdn": null,  
  "bank_account_number": null,  
  "swift_bic_code": null,  
  "iban": "DE05975683053163247100"  
},  
"source": {  
  "country_iso_code": "TUR",  
  "currency": "USD",  
  "amount": 1  
},  
"destination": {  
  "currency": "EUR",  
  "amount": 0.91  
},  
"payer": {  
  "id": 1809,  
  "name": "All Banks Germany / EUR / Payment System: SEPA",  
  "currency": "EUR",  
  "country_iso_code": "DEU",  
  "service": {  
    "id": 2,  
    "name": "BankAccount"  
  }  
},  
"sender": {  
  "lastname": "Smith",  
  "lastname2": null,
```

	<pre>"middlename": null, "firstname": "Smith", "nativename": null, "nationality_country_iso_code": null, "code": "code placeholder...", "date_of_birth": "1980-12-19", "country_of_birth_iso_code": "TUR", "gender": null, "address": "12B Baker Street", "postal_code": null, "city": "Amsterdam", "country_iso_code": "TUR", "msisdn": null, "email": null, "id_type": null, "id_country_iso_code": null, "id_number": null, "id_delivery_date": null, "id_expiration_date": null, "occupation": null, "province_state": null, "beneficiary_relationship": null, "source_of_funds": null, "bank_account_number": null }, "beneficiary": { "lastname": "Anderson", "lastname2": null,</pre>
--	---

	<pre>"middlename": null, "firstname": "Anderson", "nativename": null, "nationality_country_iso_code": null, "code": null, "date_of_birth": null, "country_of_birth_iso_code": null, "gender": null, "address": null, "postal_code": null, "city": null, "country_iso_code": "TUR", "msisdn": null, "email": null, "id_type": null, "id_country_iso_code": null, "id_number": null, "id_delivery_date": null, "id_expiration_date": null, "occupation": null, "bank_account_holder_name": null, "province_state": null }, "callback_url": null, "sent_amount": { "currency": "USD", "amount": 1 },</pre>
--	--

	<pre> "wholesale_fx_rate": 0.905999756119403, "retail_rate": null, "retail_fee": 0, "retail_fee_currency": null, "fee": { "currency": "USD", "amount": 1.5 }, "purpose_of_remittance": "ADVISORY_FEES", "document_reference_number": "ABC1234-9876", "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "reference": null, "receiving_business": null, "sending_business": null } </pre>
404	<pre> Failure [{ "code": "1008004", "message": "Transaction not found" }] </pre>

21. Get Transaction by invoice Id

*By Calling this end point we can get the transaction details by the **invoice id**.
In the query parameter we have to pass the valid **invoice id**(e.g 4368838).*

Content Type

application/json

Request

Method	URL
GET	Test https://imt.sipay.com.tr /v2/money-transfer/transactions/ext-{invoice_id}

Response

200	Success { "id": 4368838, "status": "70000", "status_message": "COMPLETED", "status_class": "7", "status_class_message": "COMPLETED", "external_id": "12001C2C", "external_code": null, "transaction_type": "C2C", "payer_transaction_reference": "1723791907", "payer_transaction_code": null, }
-----	---

```
"creation_date": "2024-08-16T07:05:03Z",
"expiration_date": "2024-08-16T08:05:03Z",
"credit_party_identifier": {
  "msisdn": null,
  "bank_account_number": null,
  "swift_bic_code": null,
  "iban": "DE05975683053163247100"
},
"source": {
  "country_iso_code": "TUR",
  "currency": "USD",
  "amount": 1
},
"destination": {
  "currency": "EUR",
  "amount": 0.91
},
"payer": {
  "id": 1809,
  "name": "All Banks Germany / EUR / Payment System: SEPA",
  "currency": "EUR",
  "country_iso_code": "DEU",
  "service": {
    "id": 2,
    "name": "BankAccount"
  }
},
"sender": {
```

```
"lastname": "Smith",  
"lastname2": null,  
"middlename": null,  
"firstname": "Smith",  
"nativename": null,  
"nationality_country_iso_code": null,  
"code": "code placeholder...",  
"date_of_birth": "1980-12-19",  
"country_of_birth_iso_code": "TUR",  
"gender": null,  
"address": "12B Baker Street",  
"postal_code": null,  
"city": "Amsterdam",  
"country_iso_code": "TUR",  
"msisdn": null,  
"email": null,  
"id_type": null,  
"id_country_iso_code": null,  
"id_number": null,  
"id_delivery_date": null,  
"id_expiration_date": null,  
"occupation": null,  
"province_state": null,  
"beneficiary_relationship": null,  
"source_of_funds": null,  
"bank_account_number": null  
},  
"beneficiary": {
```


	<pre>"lastname": "Anderson", "lastname2": null, "middlename": null, "firstname": "Anderson", "nativename": null, "nationality_country_iso_code": null, "code": null, "date_of_birth": null, "country_of_birth_iso_code": null, "gender": null, "address": null, "postal_code": null, "city": null, "country_iso_code": "TUR", "msisdn": null, "email": null, "id_type": null, "id_country_iso_code": null, "id_number": null, "id_delivery_date": null, "id_expiration_date": null, "occupation": null, "bank_account_holder_name": null, "province_state": null }, "callback_url": null, "sent_amount": { "currency": "USD",</pre>
--	---

	<pre> "amount": 1 }, "wholesale_fx_rate": 0.905999756119403, "retail_rate": null, "retail_fee": 0, "retail_fee_currency": null, "fee": { "currency": "USD", "amount": 1.5 }, "purpose_of_remittance": "ADVISORY_FEES", "document_reference_number": "ABC1234-9876", "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "reference": null, "receiving_business": null, "sending_business": null } </pre>
404	<pre> Failure [{ "code": "1008004", "message": "Transaction not found" }] </pre>

22. Get Confirm Transaction by invoice Id

*By Calling this end point we can get the transaction details by the **invoice id**.
In the query parameter we have to pass the valid **invoice id**(e.g 4368838).*

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/transactions/ext-{invoice_id}/confirm

Response

200	Success { "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "callback_url": null, "creation_date": "2024-08-16T09:46:08Z", "credit_party_identfier": { "iban": "DE05975683053163247100" }, }
-----	---

```
"destination": {
  "amount": 0.91,
  "currency": "EUR"
},
"document_reference_number": "ABC1234-9876",
"expiration_date": "2024-08-16T10:46:07Z",
"external_code": null,
"invoice_id": "1214C2C",
"external_id": "1214C2C",
"fee": {
  "amount": 1.5,
  "currency": "USD"
},
"id": 4369047,
"payer": {
  "country_iso_code": "DEU",
  "currency": "EUR",
  "id": 1809,
  "name": "All Banks Germany / EUR / Payment System: SEPA",
  "service": {
    "id": 2,
    "name": "BankAccount"
  }
},
"payer_transaction_code": null,
"payer_transaction_reference": null,
"purpose_of_remittance": "ADVISORY_FEES",
"receiving_business": null,
```

	<pre> "reference": null, "retail_fee": null, "retail_fee_currency": null, "retail_rate": null, "sending_business": null, "sent_amount": { "amount": 1, "currency": "USD" }, "source": { "amount": 1, "country_iso_code": "TUR", "currency": "USD" }, "status": "20000", "status_class": "2", "status_class_message": "CONFIRMED", "status_message": "CONFIRMED", "transaction_type": "C2C", "wholesale_fx_rate": 0.905899562885572 } </pre>
404	<p>Failure</p> <pre> [{ "code": "1008004", "message": "Transaction not found" }] </pre>

	1
--	---

23. Get Confirm Transaction by transaction Id

*By Calling this end point we can get the transaction details by the **transaction id**.
In the query parameter we have to pass the valid **transaction id**(e.g 4368838).*

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/money-transfer/transactions/{id}/confirm

Response

200	Success { "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "callback_url": null, "creation_date": "2024-08-16T09:46:08Z", "credit_party_identfier": {
-----	--

	<pre>"iban": "DE05975683053163247100" }, "destination": { "amount": 0.91, "currency": "EUR" }, "document_reference_number": "ABC1234-9876", "expiration_date": "2024-08-16T10:46:07Z", "external_code": null, "invoice_id": "1214C2C", "external_id": "1214C2C", "fee": { "amount": 1.5, "currency": "USD" }, "id": 4369047, "payer": { "country_iso_code": "DEU", "currency": "EUR", "id": 1809, "name": "All Banks Germany / EUR / Payment System: SEPA", "service": { "id": 2, "name": "BankAccount" } }, "payer_transaction_code": null, "payer_transaction_reference": null,</pre>
--	--

	<pre> "purpose_of_remittance": "ADVISORY_FEES", "receiving_business": null, "reference": null, "retail_fee": null, "retail_fee_currency": null, "retail_rate": null, "sending_business": null, "sent_amount": { "amount": 1, "currency": "USD" }, "source": { "amount": 1, "country_iso_code": "TUR", "currency": "USD" }, "status": "20000", "status_class": "2", "status_class_message": "CONFIRMED", "status_message": "CONFIRMED", "transaction_type": "C2C", "wholesale_fx_rate": 0.905899562885572 } </pre>
404	<pre> Failure [{ "code": "1008004", </pre>

	<pre> "message": "Transaction not found" }] </pre>
--	---

24. Send Money

By Calling this end point we will be able to send money and complete the full transaction cycle in a single click

Content Type

application/json

Request

Method	URL
POST	Test https://imt.sipay.com.tr/v2/send-money

Response

200	Success <pre> { "additional_information_1": null, "additional_information_2": null, "additional_information_3": null, "callback_url": null, </pre>
-----	--

```
"creation_date": "2024-08-16T09:46:08Z",

"credit_party_identifiers": {

  "iban": "DE05975683053163247100"

},

"destination": {

  "amount": 0.91,

  "currency": "EUR"

},

"document_reference_number": "ABC1234-9876",

"expiration_date": "2024-08-16T10:46:07Z",

"external_code": null,

"invoice_id": "1214C2C",

"external_id": "1214C2C",

"fee": {

  "amount": 1.5,

  "currency": "USD"

},

"id": 4369047,

"payer": {

  "country_iso_code": "DEU",

  "currency": "EUR",

  "id": 1809,

  "name": "All Banks Germany / EUR / Payment System: SEPA",

  "service": {

    "id": 2,

    "name": "BankAccount"

  }

},
```

	<pre> "payer_transaction_code": null, "payer_transaction_reference": null, "purpose_of_remittance": "ADVISORY_FEES", "receiving_business": null, "reference": null, "retail_fee": null, "retail_fee_currency": null, "retail_rate": null, "sending_business": null, "sent_amount": { "amount": 1, "currency": "USD" }, "source": { "amount": 1, "country_iso_code": "TUR", "currency": "USD" }, "status": "20000", "status_class": "2", "status_class_message": "CONFIRMED", "status_message": "CONFIRMED", "transaction_type": "C2C", "wholesale_fx_rate": 0.905899562885572 } </pre>
404	<p>Failure</p> <p>[</p>

	<pre>{ "code": "1008004", "message": "Transaction not found" }]</pre>
--	--

HTTP Codes

These are the generic server status codes.

HTTP Code	Description
200	OK
201	Created
202	Accepted (Request accepted, and queued for execution)
400	Bad request
401	Authentication failure
403	Forbidden

404	Resource not found
405	Method Not Allowed
409	Conflict
412	Precondition Failed
413	Request Entity Too Large
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

Application/Status Codes:

- ✓ 100 indicates a successful operation.

[10] Requests

Status Code (Integer)	Description
1001	Basic Validation Error
1002	IP Validation Error
1003	Hash Validation Error

1004	Currency Not Supported
1005	Invalid Total Amount
1012	Invalid Character

[20] Identification

Status Code (Integer)	Description
2001	User Not Found.
2002	Merchant Not Found.
2003	Sub Merchant Not Found
2004	Receiver Not Found
2005	Wallet Not found
2006	Currency Not Found
2007	Currency Setting Not Found
2008	Amount can't be less than
2009	Add Wallet Error. Already Exists.
2010	Delete Wallet Error. Transactions Pending.
2011	Insufficient Balacne.
2018	Merchant Key Identification Error
2023	Wallet Not Found For Currency
2024	Pos Not Set
2025	Merchant Authorization Error

[30] Sale

Status Code (Integer)	Description
3001	Sale Not Found.
3002	Merchant Pos Commission Is Not Setup For Transaction
3003	Pos Not Found
3004	Receiver Not Found
3005	Bank Not Found For Transaction
3006	Transaction Already In Progress
3007	Record Insertion Fail
3008	Payment Failed

[40] Refund

Status Code (Integer)	Description
4001	Non Refundable Transaction State
4002	Refund Amount Exceeded The Transaction Total
4003	Back to Back Refund Not Allowed Within Grace Period
4004	Receiver Not Found
4005	Duplicate Refund Reference Id

4006	Payment type is not in payment received options for refund
4007	Record Insertion Fail
4008	Refund Failed
4010	Refund request error due to unprocessed transaction

[90] General

Status Code (Integer)	Description
9001	General Error
9002	Backend Failure
9003	Unexpected Response Failure