

DTU API for EVRS

This document contains the details of the API endpoint for the direct top up to the EVRS redemption system, or DTU (Direct Top-up).

Version Control

Current document status: Draft
Current document version: 1.0.0

<i>Revision</i>	<i>Changes</i>	<i>Dated</i>
1.0.0	Initial creation of documentation (Draft)	2024-06-21
1.0.1	Update documentation to refer to publication/software	2024-06-24
1.0.2	Get Balance update to handle multiple providers	2024-06-25
1.0.3	Get balance for all providers and publications	2024-06-28

Contents

DTU API for EVRS	1
Version Control	1
Environments.....	3
Authentication & Requirements.....	4
Responses	5
Successful response structure	5
Pagination	6
Data Retrieval.....	7
Retrieve Publishers	8
Retrieve All Denominations	9
Retrieve All Denominations For Publication.....	11
Transaction Retrieval	13
Get Transactions By Publication	14
Get Transactions By Date.....	16
Get Transactions By Reference ID.....	18
Account	19
Retrieve account balance.....	20
Direct Top Up	22
Check User	23
Top Up Account.....	24
Error Codes	26

Environments

Each environment is distinguished by its own, unique domain name (FQDN). User accounts from one domain will not work on another environment

Environment	Abbreviation	Description
Production	Production	The production, live environment.
Quality Assurance / Sandbox	Qa	The quality assurance and sandbox / testing environment for external integration testing.

Authentication & Requirements

Authentication with the API endpoint requires that the authentication credentials be passed as post parameters in every call. Each call will require the following fields by default:

Note: All field keys are case sensitive.

- username
- password
- public_key

The credentials can be obtained from the API server's system administrator.

Example of x-www-form-urlencoded:

```
username: TestUser 01  
password: 46854SSfgGASG54654aas#2  
public_key: VW7lqegsLSHdPjxRWdrlQnJgQ3CzhS5f
```

These credentials will not work in any environment.

All integrations, be it to the production or sandbox environment, requires IPs to be whitelisted.

Responses

Successful response structure

Successful responses from the API endpoints will follow this structure:

```
{
  "success": true,
  "request_uri": "/publishers",
  "environment": "qa",
  "started": "2024-06-21 10:05:00",
  "reference_id": 393,
  "execution_time_ms": 11,
  "result": [ ... ]
}
```

Field	Value Type	Description
success	Boolean	If the request completed successfully on the requested server.
request_uri	String	The endpoint URI that was called.
environment	String	The environment name of the server called. Production environment will return 'production'.
started	String	Date and time in UTC when the call initially started on the server.
reference_id	Int	Endpoint call reference ID.
execution_time_ms	Int	Time in milliseconds that the server took to process the request.
result	Array	The full result of your request.

Pagination

Some of the calls to the API server can contain a lot of information, and as a result, pagination was incorporated into these calls. There is a standard of twenty (20) results to be shown on these results. Retrieving more information than the default twenty, will require that a page number be passed. To switch between result pages, pass the following field:

- page

Note: The page field must contain a valid number. This is limited to one (1) and higher. Any other field value passed, will result in an error being returned.

Data Retrieval

Data from the API server can be retrieved with the following endpoints:

Name	Endpoint
Retrieve Publications	/publications
Retrieve Denominations	/denominations
Retrieve Publication's Denominations	/denominations/{{ id }}

Retrieve Publishers

`/publications`

Retrieve a list of all publishers assigned to the consumer account.

Pagination: No

Parameters: NONE

Successful response example:

```
{
  "success": true,
  "request_uri": "/publications",
  "environment": "qa",
  "started": "2024-06-21 10:05:00",
  "reference_id": 393,
  "execution_time_ms": 11,
  "result": [
    {
      "publisher_id": 6,
      "publisher_name": "AfterVerse",
      "publication_id": 5,
      "publication_name": "PK XD"
    },
    {
      "publisher_id": 5,
      "publisher_name": "Creative Monkey",
      "publication_id": 4,
      "publication_name": "Cricket Gangsta"
    },
    {
      "publisher_id": 3,
      "publisher_name": "EA Games",
      "publication_id": 2,
      "publication_name": "Fifa"
    }
  ]
}
```

Details:

Field	Value Type	Description
publisher_id	int	The system ID for the software publisher.
publisher_name	string	The system name for the software publisher.
publication_id	Int	The ID of the publication.
publication_name	string	The name of the publication.

Retrieve All Denominations

[/denominations](#)

Retrieve a list of all denominations for all the publication assigned to the consumer account.

Pagination: No

Parameters: NONE

Successful response example:

```
{
  "success": true,
  "request_uri": "/denominations",
  "environment": "qa",
  "started": "2024-06-21 11:08:34",
  "reference_id": 395,
  "execution_time_ms": 12,
  "result": [
    {
      "id": 18,
      "publisher_id": 3,
      "publisher_name": "EA Games",
      "publication_id": 2,
      "publication_name": "EA FC Mobile",
      "region_id": 334,
      "region_name": "EA FC Singapore",
      "region_locked": true,
      "country_name": "Singapore",
      "country_code_2": "SG",
      "country_code_3": "SGP",
      "vat_enabled": true,
      "currency_symbol": "ARS",
      "currency_name": "Argentine Peso",
      "denomination_name": "EA FC MOBILE 999 Silver SGP",
      "denomination_face_value": "14.98",
      "last_updated": "2024-06-21 11:00:01",
      "denomination_dtu_enabled": true
    },
    { ... }
  ]
}
```

Details:

Field	Value Type	Description
id	Int	The denomination ID.
publisher_id	Int	ID of the software publisher.
publisher_name	String	The name of the software publisher.
publication_id	Int	ID of the publication the denomination is associated with.
publication_name	String	The name of the publication (software).
region_id	Int	ID of the region of the denomination.

Direct Top-up API Documentation

region_name	String	Denomination regional name.
region_locked	Boolean	If the denomination is locked to its set region.
country_name	String	Name of the country of the denomination.
country_code_2	String	ISO 2 country code for the denomination.
country_code_3	String	ISO 3 country code for the denomination.
vat_enabled	Boolean	If the denomination has vat associated with it.
currency_symbol	String	The symbol of the denomination's face value.
currency_name	String	The name of the currency of the denomination's face value.
denomination_name	String	Name of the denomination.
denomination_face_value	String	Face value of the denomination.
last_updated	String	Date that the denomination was last updated by the system.
denomination_dtu_enabled	Boolean	If the denomination has been enabled for direct top up via API for the voucher provider.

Retrieve All Denominations For Publication

`/denominations/{id}`

Retrieve a list of all denominations for a specific publication, passing the system `publication_id` in the URI.

Pagination: No

Parameters: NONE

Successful response example:

```
{
  "success": true,
  "request_uri": "/denominations/5",
  "environment": "qa",
  "started": "2024-06-21 11:26:51",
  "reference_id": 396,
  "execution_time_ms": 12,
  "result": [
    {
      "id": 32,
      "publisher_id": 6,
      "publisher_name": "AfterVerse",
      "publication_id": 5,
      "publication_name": "PK XD",
      "region_id": 376,
      "region_name": "PK XD India",
      "region_locked": true,
      "country_name": "India",
      "country_code_2": "IN",
      "country_code_3": "IND",
      "vat_enabled": true,
      "currency_symbol": "USD",
      "currency_name": "United States Dollar",
      "denomination_name": "PKXD 1250 GEMS",
      "denomination_face_value": "29.99",
      "last_updated": "2024-06-21 11:00:01",
      "denomination_dtu_enabled": true
    },
    { ... }
  ]
}
```

Details:

Field	Value Type	Description
id	Int	The denomination ID.
publisher_id	Int	ID of the software publisher.
publisher_name	String	The name of the software publisher.
publication_id	Int	ID of the publication the denomination is associated with.
publication_name	String	The name of the publication.

Direct Top-up API Documentation

region_id	Int	ID of the region of the denomination.
region_name	String	Denomination regional name.
region_locked	Boolean	If the denomination is locked to its set region.
country_name	String	Name of the country of the denomination.
country_code_2	String	ISO 2 country code for the denomination.
country_code_3	String	ISO 3 country code for the denomination.
vat_enabled	Boolean	If the denomination has vat associated with it.
currency_symbol	String	The symbol of the denomination's face value.
currency_name	String	The name of the currency of the denomination's face value.
denomination_name	String	Name of the denomination.
denomination_face_value	String	Face value of the denomination.
last_updated	String	Date that the denomination was last updated by the system.
denomination_dtu_enabled	Boolean	If the denomination has been enabled for direct top up via API for the voucher provider.

Transaction Retrieval

Monetary transactions can be retrieved from the following endpoints:

Name	Endpoint
Get Transactions By Publication	/transactions/publication
Get Transactions By Date	/transactions/date
Get Transactions By Reference ID	/transactions/reference

Get Transactions By Publication

</transactions/publication>

Retrieve a list of all monetary transactions that have been completed by the current consumer for the specified publication.

Pagination: Yes

Parameters:

Field	Value Type	Description	Required
publication_id	int	The system ID of the publication (software) to retrieve all monetary transactions for	Yes
page	int	The page to retrieve (defaults to 1)	No
start_date	string	The start date to retrieve transactions from, in the year, month, day format (YYYY-MM-DD)	Yes
end_date	string	The end date to retrieve transaction to, in the year, month, day format (YYYY-MM-DD)	No
limit	Int	The amount of records to return per call. Maximum (server-controlled) value is 200.	No

Successful response example:

```
{
  "success": true,
  "request_uri": "/transactions/publication",
  "environment": "qa",
  "started": "2024-06-21 11:37:15",
  "reference_id": 398,
  "execution_time_ms": 14,
  "result": [
    {
      "system_id": 356,
      "reference": "862493117",
      "result": {
        "message": "Denomination topped up successfully.",
        "publication_id": 2,
        "publication_name": "EA FC Mobile",
        "denomination_id": 3,
        "denomination_name": "EA FC 520 Points MYS",
        "player_id": "851276981637005312",
        "top_up_details": {
          "voucher_id": 42,
          "provider_item_id": 19798809,
          "purchase_transaction_id": 11217836,
          "financial_reference": "24189849",
          "sales_reference": 18028486,
          "serial_number": "E3FEBC1D-000004335"
        }
      }
    }
  ],
}
```

Direct Top-up API Documentation

```
{...}  
]  
}
```

Details:

The result will contain the output of every monetary transaction made between the two date ranges.

NOTES:

- If the end date is not specified, the date range will be from the start to the end of the start_date parameter.
- Each result will return the response originally received

Get Transactions By Date

</transactions/date>

Retrieve a list of all monetary transactions that have been completed by the date range supplied.

Pagination: Yes

Parameters:

Field	Value Type	Description	Required
page	int	The page to retrieve (defaults to 1)	No
start_date	string	The start date to retrieve transactions from, in the year, month, day format (YYYY-MM-DD)	Yes
end_date	string	The end date to retrieve transaction to, in the year, month, day format (YYYY-MM-DD)	No
limit	int	The amount of records to return per call. Maximum (server-controlled) value is 200.	No

Successful response example:

```
{
  "success": true,
  "request_uri": "/transactions/date",
  "environment": "qa",
  "started": "2024-06-21 11:41:41",
  "reference_id": 399,
  "execution_time_ms": 17,
  "result": [
    {
      "system_id": 371,
      "reference": "994929260",
      "result": {
        "role_id": 536,
        "message": "Denomination topped up successfully.",
        "publication_id": 5,
        "publication_name": "PK XD",
        "denomination_id": 29,
        "denomination_name": "PKXD 50 GEMS",
        "player_id": "maluquito#720",
        "transaction_date": "2024-06-14 20:15:56",
        "top_up_details": {
          "voucher_id": 51,
          "provider_item_id": 19798923,
          "purchase_transaction_id": 11217847,
          "financial_reference": "24189858",
          "sales_reference": 18028495,
          "serial_number": "8E007D92-000004449"
        }
      }
    },
    { ... }
  ]
}
```


Direct Top-up API Documentation

Details:

The result will contain the output of every monetary transaction made between the two date ranges.

NOTES:

- If the end date is not specified, the date range will be from the start to the end of the start_date parameter.
- Each result will return the response originally received

Get Transactions By Reference ID

/transactions/reference

Retrieve a list of all monetary transactions that have been completed using the supplied reference ID.

Pagination: Yes

Successful response example:

```
{
  "success": true,
  "request_uri": "/transactions/reference",
  "environment": "qa",
  "started": "2024-06-21 11:43:26",
  "reference_id": 400,
  "execution_time_ms": 13,
  "result": [
    {
      "system_id": 291,
      "reference": "2780345741",
      "result": {
        "message": "Denomination topped up successfully.",
        "publication_id": 5,
        "publication_name": "PK XD",
        "denomination_id": 29,
        "denomination_name": "PKXD 50 GEMS",
        "player_id": "maluquito#720",
        "top_up_details": {
          "voucher_id": 36,
          "provider_item_id": 19798917,
          "purchase_transaction_id": 11217824,
          "financial_reference": "24189838",
          "sales_reference": 18028475,
          "serial_number": "5E052C7D-000004443"
        }
      }
    },
    { ... }
  ]
}
```

Details:

The result will contain the output of every monetary transaction made using the reference ID.

NOTES:

- Each result will return the response originally received

Account

Current consumer account details can be retrieved via the following endpoints:

Name	Endpoint
Get Account Balance	/balance

Retrieve account balance

/balance

Retrieve the consumer balance for the specified publication.
Each voucher provider will be returned for the current account.
Each provider will list the publications that is assigned to the provider.

Pagination: No

Parameters:

NONE

Successful response example:

```
{
  "success": true,
  "request_uri": "/balance",
  "environment": "local",
  "started": "2024-06-28 12:08:39",
  "reference_id": 799,
  "execution_time_ms": 651,
  "result": [
    {
      "provider_id": 7,
      "provider_name": "VVS Demo",
      "currency": "USD",
      "balance": 1047328.32,
      "provider_publications": [
        {
          "publication_id": 4,
          "publication_name": "Cricket Gangsta"
        },
        {
          "publication_id": 2,
          "publication_name": "Fifa"
        },
        {
          "publication_id": 5,
          "publication_name": "PK XD"
        }
      ]
    }
  ]
}
```

Details:

Field	Value Type	Description
provider_id	Int	DTU provider ID.
provider_name	String	Name of the DTU provider.
currency	String	ISO code for the wallet currency.
balance	Decimal	Account balance.

Direct Top-up API Documentation

provider_publications	Array	A list of all publications that use the same provider.
-----------------------	-------	--

Direct Top Up

Directly top up a player account and check player account details via the following endpoints:

Name	Endpoint
Check User	/user/check
Top Up Account	/product/redeem

Check User

[/user/check](#)

Check the player's account details, validity and retrieve all available denominations for that player.

Pagination: No

Parameters:

Field	Value Type	Description	Required
reference_id	String	A unique reference for the transaction on the user check call to the publication/software provider. UUID is preferred.	Yes
publication_id	Int	System ID of the publication.	Yes
player_id	String	The player ID of the account to check.	Yes

Successful response example:

```
{
  "success": true,
  "request_uri": "/user/check",
  "environment": "qa",
  "started": "2024-06-21 12:22:00",
  "reference_id": 407,
  "execution_time_ms": 601,
  "result": {
    "user_id": 536,
    "available_denominations": 4,
    "available_denomination_records": [
      { ... }
    ]
  }
}
```

Details:

Field	Value Type	Description
user_id	Int	System user ID for the requested player.
available_denominations	Int	The amount of denominations that are valid for the player account.
available_denomination_records	Array	A list of denominations available for the player account. Structure as here

Top Up Account

</product/redeem>

Redeem a denomination directly to a player account.

Pagination: No

Parameters:

Field	Value Type	Description	Required
reference_id	String	A unique reference for the top up request.	Yes
denomination_id	Int	The system ID for the denomination to top the account up with.	Yes
publication_id	Int	System ID of the publication/software the account is for.	Yes
player_id	String	The player name / account to top up.	Yes

Successful response example:

```
{
  "success": true,
  "request_uri": "/product/redeem",
  "environment": "qa",
  "started": "2024-06-21 12:16:57",
  "reference_id": 403,
  "execution_time_ms": 3863,
  "result": {
    "role_id": 536,
    "message": "Denomination topped up successfully.",
    "publication_id": 5,
    "publication_name": "PK XD",
    "denomination_id": 29,
    "denomination_name": "PKXD 50 GEMS",
    "player_id": "maluquito#720",
    "transaction_date": "2024-06-21 12:17:01",
    "top_up_details": {
      "voucher_id": 53,
      "provider_item_id": 19798971,
      "purchase_transaction_id": 11218065,
      "financial_reference": "24189912",
      "sales_reference": 18028553,
      "serial_number": "30948AA0-000004497"
    }
  }
}
```

Details:

Field	Value Type	Description
role_id	Int	The system role ID assigned to the redemption / top up.
message	String	Server response message of the action.
publication_id	Int	System ID of the publication of the transaction.

Direct Top-up API Documentation

publication_name	String	System name of the publication of the transaction.
denomination_id	Int	System ID of the denomination of the transaction.
denomination_name	String	System denomination name.
player_id	String	The player ID of the transaction.
transaction_date	String	The date and time in UTC of the transaction.
top_up_details	Array	Top Up Details

“top_up_details” Details

Field	Value Type	Description
voucher_id	Int	System ID of the voucher associated with the transaction.
provider_item_id	Int	System ID returned from the voucher code provider.
purchase_transaction_id	Int	System ID returned from the voucher code provider.
financial_reference	Int	Financial reference ID for the transaction returned from the voucher code provider.
sales_reference	Int	Sales reference ID as returned from the voucher code provider.
serial_number	String	Serial number of the top up transaction as returned from the voucher provider.

Error Codes

A list of error codes and meaning that can be returned by the API call.

Code	Message
1	Network failed
2	Parse string failed
3	Network failed
4	Query signing failed
5	Network failed
6	JSON decode failed
7	Response != 200
8	Exception occurred
9	Response != 200
10	JSON decode failed
11	API request parameters incorrect or not supported
12	Invalid request
13	Incorrect request format
14	Failed to resolve request
15	Missing sCloudApiName parameter
16	Invalid sCloudApiName parameter or backend API not configured
17	Authentication failed. See the error description for details.
18	Authentication failed. The authentication API returns a value greater than 0
19	Incorrect sCloudApiName parameter or incorrect backend API configuration
20	Invalid request
21	Failed to call backend http service
22	Server busy
23	Calling frequency is too high - request rate expansion
24	Prohibited calls: calling private network APIs from public network
25	No valid serial number parameter is passed
26	Network exception of calling backend service, such as timeout
27	Internal error, memory allocation failed
28	Incorrect parameter of anti-brush request
29	Incorrect parameters for calling IDC API
30	Incorrect sCloudApiName parameter or incorrect backend API configuration
31	Invalid UIN parameter
32	Failed to get the caller source host ip, failed to convert source ip to ipv6
33	Backend service timed out
34	Invalid appid parameter [1]
35	Invalid skey parameter [1]
36	Invalid appid parameter [2]
37	Invalid skey parameter [2]

Direct Top-up API Documentation

38	Invalid openid parameter [1]
39	Invalid appid parameter [3]
40	Invalid openid parameter [2]
41	Invalid access_token parameter
42	Invalid sAMSSourceAppId parameter
43	Invalid sAMSAppOpenId parameter
44	Invalid sAMSAccessToken parameter
45	Invalid sAMSTargetAppId parameter
46	Internal error, error encrypting data
47	Incorrect sCloudApiName parameter or background API configuration does not exist
48	Incorrect sCloudApiName parameter or incorrect backend API configuration
49	Service capacity exceeded
50	Protocol not supported
51	Signature error
52	An unknown error has occurred
53	Player ID supplied does not exist for this game and user
54	Player ID verification failed
55	Voucher status fail / not redeemable
56	Player role update failed
57	Player ID does not match a valid APEX account
58	Player account login banned
59	API result != OK
60	An unconfigured API function has been called
61	Unmatched game id
62	Exception occurred
63	Exception occurred
64	Empty results
65	API data response result != 0
66	API result != OK
67	API data response result != 0
68	Check PIN failed
69	Invalid player role details
70	Empty Voucher ID - topup interrupted
71	Voucher and Player regions mismatch
72	Player is not available.
73	Unable to process the request.
74	Requirement not satisfied.
75	Duplicate receipt.
76	Player is underaged
77	Player is banned.
78	Product mapping not enabled for provider.

Direct Top-up API Documentation

80	Could not interpret received JSON response.
79	Out of region voucher. Player in :PLAYER_COUNTRY_ISO voucher in :VOUCHER_COUNTRY_ISO
81	Could not determine redeemable state of specified voucher.
82	Voucher found for specified PIN is not in correct state.
83	Could not retrieve voucher record for specified PIN.
84	Your account does not have access to the requested publisher.
85	Error retrieving status of specified voucher.
86	Your account does not have access to the requested publisher.
87	More than one voucher found for current query.
88	No product record found for current game reference.
89	Integration reference could not be found.
90	Invalid PIN for selected game.
91	Voucher is in a non-verifiable state.
92	Game provider could not be reached.
93	Game provider internal error occurred.
94	Invalid Auth token received from provider.
95	Auth token verification failed from provider.
96	Request data missing error received from provider.
97	Internal error from provider.
98	Invalid data error from provider.
99	Result found error received from provider.
100	Result not found by provider.
101	PIN is invalid. Not generated.
102	PIN is invalid. Not stored.
103	PIN is invalid. Not distributed.
104	PIN not found in VVS database.
105	PIN corrupted: found multiple times in VVS database.
106	PIN corrupted: not in SOLD state.
107	Call to provider returned an error
108	Provider endpoint was not found.
109	Provider encountered internal error.
110	Call to provider is missing required parameters.
111	Invalid nickname format. It should be in the format: <nickname#123 (3 to 5 digits).
112	User not found.
114	Player is underaged
113	Voucher has already been redeemed.
116	User not found.
115	Please wait :TIME seconds before attempting to log in again.
117	Player is banned.
155	Transaction cool-off period is over. Transaction can no longer be reversed.
156	Could not reverse vouchers from sales transaction.

Direct Top-up API Documentation

157	API consumer record could not be found for current user.
158	API user record could not be found for current transaction.
159	Denomination game does not match the requested game.
160	API publisher record could not be found for current transaction.
152	Denomination flagged as already redeemed.
153	Reversal of transaction has failed.
154	Transaction has already been reversed.
118	No username specified.
119	No password specified.
120	User account not found.
121	Incorrect password supplied.
122	IP address not whitelisted.
123	Endpoint route is invalid.
124	No games assigned to user profile.
125	No reference_id passed on required call.
126	Passed reference_id duplicate.
127	Missing required parameters: {\$args}.
128	Page passed with no value.
129	Passed page value is not valid.
130	Could not find specified denomination.
131	Multiple denominations were found.
132	Specified denomination is no longer active.
133	Specified denomination contains invalid product information.
134	Specified denomination is not enabled for direct top up.
135	No public key specified.
136	Encryption required but no vector passed.
137	Encryption vector length is invalid. 16 characters required.
138	Internal error occurred while retrieving games list.
139	DTU provider not found.
140	Internal error occurred while retrieving denomination list.
141	Internal error occurred while retrieving game denomination list.
142	Internal error occurred while topping up player.
143	Internal error occurred retrieving transactions by reference.
144	Internal error occurred retrieving game transactions by date range.
145	Internal error occurred retrieving user transactions by date range.
146	Internal error occurred while updating denominations.
147	Internal error occurred while checking user details from provider.
148	Internal error occurred while purchasing a voucher PIN.
149	Internal error occurred while reversing voucher.
150	Voucher not in a reversable state.
151	Internal error occurred while calling reverse voucher.

Direct Top-up API Documentation

999	Unknown error
1000	Product SKU validation has failed