

**bet●tech**

# INTEGRATION GUIDE

Version 1.16 July 25, 2019

# Contents

<b>Change log</b>	<b>3</b>
<b>Terms</b>	<b>3</b>
<b>Important notes</b>	<b>5</b>
<b>Iframe integration</b>	<b>5</b>
Betslip initialization	5
Installing BTRenderer script	6
Defining container	6
Initializing	6
Method COMMAND	8
Linking	8
Change top position of Betslip	8
Clean up BT Library	8
<b>List of methods to implement on Partner side</b>	<b>9</b>
Method IDENTIFY	10
Method BET_MAKE	12
Method BET_COMMIT (optional)	18
Method BET_SETTLEMENT (optional)	19
Method BET_REFUND	20
Method BET_WIN	22
Method BET_CANCEL	25
Method BET_DISCARD	28
Method BET_ROLLBACK	28
A bet's cycle of life	30
<b>External API</b>	<b>32</b>
Method PING	33
<b>Bonus API</b>	<b>33</b>
Method TEMPLATES	33
Method PLAYER_BONUSES	39
Method BONUS	44
Method MASS_GIVE_BONUS	45

Method REVOKE_BONUS	51
<b>Appendices</b>	<b>56</b>
JWT Token example	56
A bet's cycle of life	58
Cash out	59
Types of bets	59
Language Support	62
Currency Support	63

## Change log

Version	Description
1.16	Added new parameter "selections" in the BET_SETTLEMENT request. Added new optional parameter "session_id" in the IDENTIFY request
1.15	The "minFrameHeight" parameter is no longer available in the BTRenderer script. The URL for BetSlip initialization is changed. Added new fields into the BET_MAKE request (cross_rate_euro, operation, is_quick_bet, odds)
1.14	Added a bet's life cycle in the appendices Removed the "amount" and "currency" fields in the BET_CANCEL method Removed the "amount" and "currency" fields in the BET_REFUND method Requests to External API should be signed with Public key (JWT token, RSA-encoded). Added new fields into External API requests (max_cap, event_scheduled). Added new fields into the BET_MAKE request (live, sport_name, category_name, tournament_name, competitor_name, market_name, outcome_name, scheduled). Added new method BET_ROLLBACK.

This document contains important information about integration process with the **Bet.Tech** sportsbook.

## Terms

### Sportsbook

An establishment/individual offering odds on the outcome of events, usually but not exclusively sports, for the purpose of accepting bets.

**Bet (Betslip, Wager)**

Risk money on the outcome of an event with the return proportionate to the perceived likelihood as measured by odds.

**Selection**

Entity a bet consist of.

**Bet Types - Single**

Betslip comprising only one selection.

**Bet Types - Combo**

Where more than one bet is combined, the odds accumulated and success is dependent on all selections winning.

**Bet Types - System**

A combination of dependant bets offering proportionately higher returns, often using standardised permutations of co-dependant selections.

**Bonus**

A special incentive proposed by Partner to players in order to stimulate activity.

**Comboboost**

A kind of bonus entitling player with an additional multiplier among co-dependant selections odds.

**Freebet**

A kind of bonus incorporating predefined amount of money issued to a player that can be used to make a bet.

**Standart Freebet - Stake not returned**

Deductible Freebet - this means that only the amount of money player won will be transferred to his balance. For example, placing a freebet of \$5 with the odd 3.5 player will get  $5 * 3.5 - 5 = \$12.50$

**Free Money Bet - Stake returned**

Not Deductible Freebet - player will receive the amount of money won and the freebet. For example, placing a freebet of \$5 with the odd 3.5 player will get  $5 * 3.5 = \$18.50$

**No Risk Bet - Bet Refunds**

When player makes bet, the balance should be reduced by the sum of freebet. In case the bet is lost, the sum will be transferred back to his balance. In case the bet is won, the payout will contain both the bet and won.

**Examples of payouts (for 5\$ bet and 3.5 odds)**

Type of bonus	Win	Refund	Asian win half	Asian lose half	Lose
Freebet	$5\$ \times 3.5 - 5\$ = 12.5\$$	0\$	$(2.5\$ \times 3.5 + 2.5\$ \times 1) - 5\$ = 6.25\$$	0\$	0\$
Freemoney	$5\$ \times 3.5 = 17.5\$$	5\$	$2.5\$ \times 3.5 + 2.5\$ \times 1 = 11.25\$$	$2.5\$ \times 0 + 2.5\$ \times 1 = 2.5\$$	0\$
No Risk Bet	$5\$ \times 3.5 = 17.5\$$	$5\$ \times 1 = 5\$$	$2.5\$ \times 3.5 + 2.5\$ \times 1 = 11.25\$$	5\$	5\$

## Important notes

1. Requests from Bet.Tech to Partner contain JSON object with payload, signed with Public Key (JWT token, RSA-encoded):

```
{"payload":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJmb28iOiJiYXliLCJpYXQiOiJlMTYyMzkwMjJ9.4F-lcDyBVt7zATi0LTijvKubIVZEx_qSf0Wu9-78VDk"}
```

Partner has to validate and unpack it. Please, refer to Appendices section to find the examples of JWT token generation and unpackage.

NB. It is strongly recommended to use systemd-timesyncd or ntpd service to synchronize the local system clock with a remote Network Time Protocol server.

2. Content-type for the requests is application/json.
3. All money transactions are carried out in cents (i.e. 100 = 1).
4. For the security reasons Partner is responsible for generating unique timely-restricted IP-binded one-time key to each player. The key must be passed to iframe throughout parameters.

## Iframe integration

In order to initialize Bet.Tech Sportsbook Frame Partner has to carry out both betslip and iframe integration.

### Betslip initialization

Betslip initialization is implemented throughout BTRenderer JavaScript class. Thus, it is necessary to create this object before initializing betslip.

## Installing BTRenderer script

Partner has to include BT library script to the page.

```
<script src="https://URL.sptpub.com/bt-renderer.min.js"></script>
```

Ask your account manager for URL

NB For testing environment use

```
https://ui.invisiblesport.com/bt-renderer.min.js
```

## Defining container

Then **insert a containing** element to the 'body' section.

```
<div id="bettech"></div>
```

## Initializing

After DOM is successfully loaded Partner has to init BTRenderer by calling **initialize()** method.

```
var bt = new BTRenderer().initialize({
  frameUrl: 'http://test.com',
  brand_id: '1111111111111111',
  key: frame_key,
  lang: lang,
  target: document.getElementById('bettech'),
  minFrameHeight: 700,
  maxBetSlipHeight: null,
  betSlipOffsetTop: 0,
  betslipZIndex: 999,
  onRouteChange: function() {},
  onLogin: function() {},
  onRegister: function() {},
  onSessionRefresh: function() {}
})
```

## Parameters description

Parameter	Type	Description
frameUrl	String	URI of the frame provided by Bet.Tech
brand_id	String	Unique Partner Website ID received during the onboarding process.
key	String	Unique timely-restricted IP-binded one-time key assigned to each player.

Parameter	Type	Description
lang	String	Frame language (default: en). The full list of supported languages can be reviewed in Appendices section. The abbreviation is case-sensitive.
target	HTMLElement	An element of DOM to render betslip
maxBetSlipHeight	Number	The parameter sets the maximum height of betslip opened.
betSlipOffsetTop	Number	The parameter defines the gap between window top and opened betslip.
betslipZIndex	Number	The parameter defines the index of betslip in relation to DOM elements used on Partners website. It helps to avoid overlapping conflicts of different elements.
onRouteChange	callback	Callback function provided by Partner that will be called when a player navigates across the frame
onLogin	callback	Callback function provided by Partner that will be called when a player logs in with the frame.
onRegister	callback	Callback function provided by Partner that will be called when a player registers with the frame.
onSessionRefresh	callback	Callback function provided by Partner that is responsible for generating new key and passing to the frame. This function will be called when session token is expired.
onRecharge	callback	Callback function provided by Partner that will be called when a player makes deposit with the frame.
scrollTopButtonPosition	String	The parameter defines the scroll button position in the mobile website version. There are two possible values: "left" and "right".

NB. Once a player has switched the language, signed in/out it is needed to call **initialize()** method and pass the updated parameters to it.

NB. It is strongly recommended to use betSlipOffsetTop parameter to avoid UX conflicts instead of setting up height limits of betslip.

## Method COMMAND

BTRendererr supports command method allowing Partner to perform supported list of actions within the frame.

```
bt.command('COMMAND_NAME', 'PARAMS');
```

Parameter	Type	Description
COMMAND_NAME	String	The name of command supported by Bet.Tech
PARAMS	Depending on command_name	Special parameters needed to carry out a given command.

### List of supported commands

COMMAND_NAME	PARAMS	Description
go_to_page	URL <String>	This method is used to redirect a player to a particular page of the frame passed in parameters.

## Linking

Partner can directly open a subpage inside frame by specifying

```
?bt-path=/subpage_address
```

in URL. To find out subpage link press [Shift] + [D].

## Change top position of Betslip

This method is being used to change the position of expanded betslip in relation to window top to avoid UX conflicts like undesired overlapping of website elements.

```
bt.betSlipOffsetTop(value);
```

Parameter	Type	Description
Value	Integer	The gap between window top and opened betslip measured in pixels

## Clean up BT Library

This method is being used to clean up DOM and unsubscribe from events.

```
bt.kill();
```



## List of methods to implement on Partner side

In order to perform successfully Partner is expected to implement the following list of endpoints by placing additional business logic along with the required behaviour from the Platform internal system.

Method	Type	URL
Ping	GET	/ping
Identify	GET	/identify
Bet Make	POST	/bet/make
Bet Commit (optional)	POST	/bet/commit
Bet Settlement (optional)	POST	/bet/settlement
Bet Refund	POST	/bet/refund
Bet Win	POST	/bet/win
Bet Cancel	POST	/bet/cancel
Bet Discard	POST	/bet/discard
Bet Rollback	POST	/bet/rollback

NB. In case of any internal error on Partner side we are waiting for HTTP 500 response. We repeat unsuccessful request several times with increasing time interval except for BET\_MAKE and IDENTIFY requests. The BET\_MAKE and IDENTIFY requests are valid short time (up to 15 seconds).## Method PING This method is used to verify the availability of API provided by the Partner's platform. In case the Partner's platform runs correctly, Bet.Tech expect Response 200 in the described format.

### Request

GET https://URL/ping

### Response 200

```
{
  "timestamp": 1539005526
}
```

Parameter	Type	Description
Timestamp	Long	Current timestamp as unixtime

## Method IDENTIFY

This method is used to identify player using key generated by Partner before. In case of successful player identification, Bet.Tech expects Response 200, otherwise Response 403.

Partner can use data in the fields 'ip' and 'uacode' to ensure that Bet.Tech frame is opened in the same session as the generated key.

For example, Partner can decline identification with the error "1004: IP address does not match" if IP from the parameters is different from the player's IP known on Partner's side. Practically, it would be better to allow identification, because the player's provider or VPN settings can lead to different IPs for accessing Partner's site and Bet.Tech frame.

### Request

GET: [https://URL/identify?key=player\\_access\\_key&ip=player\\_ip](https://URL/identify?key=player_access_key&ip=player_ip)

Parameter	Description
key	Player access key
ip	"Player IP, as IPv4 string"
uacode	MD5 sum of the player's User-Agent. Optional.

### Response 200

```
{
  "user_id": "1660986605374148608",
  "username": "John",
  "lang": "EN",
  "currency": "USD",
  "balance": 100000,
  "feature_flags": {
    "is_cashout_available": true
  },
  "session_id": "aa8499f578b13e9147428c30ad8c63bd"
}
```

Parameter	Type	Description
user_id	String	Unique ID of player assigned by Partner.
username	String	Parameter passed to display players username in Bet.Tech Backoffice. Any value is acceptable.
lang	String	Language code. The full list of languages supported can be found in the Appendices.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
balance	Integer	Current balance of player.
feature_flags	Object	Parameter indicating which features of Bet.Tech are available for a given player. Optional (see Feature flags)
session_id	String	Optional parameter for player identification assigned by Partner. If it is used will be referred to the BET_MAKE request.

**Feature flags** Contact your Bet.Tech Account Manager to get a full list of feature flags supported by Bet.Tech

Parameter	Type	Description
is_cashout_available	Boolean	The mark of whether cash out option should be available for a given player or not.
is_match_tracker_available	Boolean	The mark of whether Match Tracker should be shown for a player.

#### Response 403

```
{
  "code": 1001,
  "message": "Key not found"
}
```

Parameter	Type	Description
code	Integer	Special identifier for error code
message	String	Extended description of what has happened during the player identification

Error code	Message
1001	Key not found
1002	Attempt to reuse the key
1003	Key is expired
1004	IP address does not match
1005	Player is blocked
1006	Player is not found

## Method BET\_MAKE

This method is used when player makes bet. Partner is expected to process the following request and provide a response in the described format.

### Request

POST https://URL/bet/make

```
{
  "amount": 1000,
  "currency": "USD",
  "player_id": "1659297400285696000",
  "bonus_id": 1664976298830860288,
  "session_id": "aa8499f578b13e9147428c30ad8c63bd",
  "transaction":
  {
    "id": "1265023428769484821",
    "betslip_id": "1659603658884648961",
    "player_id": "1659299365132570624",
    "operator_id": "1657013002915142201",
    "operator_brand_id": "1653815133341880320",
    "ext_player_id": "1659297400285696000",
    "timestamp": 1538654560.2354896,
    "status": 0,
    "amount": 1000,
    "currency": "USD",
    "cross_rate_euro": "0.893598",
    "operation": "bet",
    "bonus_id": 1664976298830860288
  },
  "betslip":
  {
    "id": "1659603658884648961",
    "timestamp": 1538654560.2215934,
    "player_id": "1659299365132570624",
    "operator_id": "1657013002915142201",
    "operator_brand_id": "1653815133341880320",
    "ext_player_id": "1659297400285696000",
    "currency": "USD",
    "type": "1/1",
```

```

"sum": 1000,
"k": "1.350",
"paid_for": false,
"paid_out": false,
"calculated": false,
"is_quick_bet": false,
"bets":
[
{
  "id": "1664727723865673728",
  "live": True
  "sport_id": "1",
  "event_id": "1664727095525380096",
  "tournament_id": "1658527741017985024",
  "category_id": "1658527740917321728",
  "sport_name": "Soccer",
  "category_name": "International Clubs",
  "tournament_name": "UEFA Champions League",
  "competitor_name": [
    "Liverpool FC",
    "FC Barcelona"
  ],
  "market_name": "1x2",
  "outcome_name": "Liverpool FC",
  "scheduled": 1557255600,
  "odds": "1.35"
}
]
}

```

Parameter	Type	Description
amount	Integer	Amount of money to be deducted from the player`s balance after the bet is being made.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
player_id	String	Unique identifier assigned to a player on Partner side.
bonus_id	Integer	Optional. Unique identifier of bonus associated with the transaction.
session_id	String	Optional. Unique player session id provided by Partner in the IDENTIFY response.
transaction	<TransactionItem>	Dictionary containing information about the transaction.
betslip	<BetSliptem>	Dictionary containing information about player`s betslip.

#### <TransactionItem>

Parameter	Type	Description
id	String	Unique identifier for the transaction assigned by Bet.Tech
betslip_id	Integer	Unique identifier of betslip assigned by Bet.Tech when bet is made.
player_id	Integer	Unique identifier of a player assigned by Bet.Tech.
operator_id	Integer	Partner unique identifier.
operator_brand_id	Integer	Partner Website unique identifier.
ext_player_id	String	Unique identifier assigned to a player on Partner side.
timestamp	Float	Transaction timestamp in unixtime.
status	Integer	Bet.Tech internal status for the transaction described below.
amount	Integer	The sum of money to be deducted from players balance expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
cross_rate_euro	String	Player's currency exchange rate to EUR.
operation	String	Type of transaction described below.
bonus_id	Integer	Optional. Unique identifier of bonus associated with the transaction.

**<BetSlipItem>**

Parameter	Type	Description
id	Integer	Unique identifier of betslip assigned by Bet.Tech when bet is made.
timestamp	Float	Betslip timestamp in unixtime.
player_id	Integer	Unique identifier of player assigned by Bet.Tech.
operator_id	Integer	Partner unique identifier.
operator_brand_id	Integer	Partner Website unique identifier.
ext_player_id	String	Unique identifier assigned to a player on Partner side.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.

Parameter	Type	Description
type	String	Abbreviation of bet type. For more information see Type of bets chapter.
sum	Integer	The amount of bet placed by player.
k	Float	Total odd of the betslip resulted by multiplying odds of all the selections.
paid_for	Boolean	The mark of either betslip has been accepted (ie the balance was decreased) or not.
paid_out	Boolean	The mark of either betslip has been won or not.
calculated	Boolean	The mark of either betslip has been settled or not.
is_quick_bet	Boolean	The mark of either betslip has been made using Quick Bet function.
bets	List of <BetItem>	The array of dictionaries containing information about each selection included in a given betslip.

**<BetItem>**

Parameter	Type	Description
id	Integer	Unique identifier of a betslip`s selection assigned by Bet.Tech
sport_id	Integer	Unique identifier of the sport.
event_id	Integer	Unique identifier of the event.
tournament_id	Integer	Unique identifier of the tournament.
category_id	Integer	Unique identifier of the category.
live	Boolean	Optional. For live maket is true, for prematch is false
sport_name	String	Optional. Name of the sport.
category_name	String	Optional. Name of the category.
tournament_name	String	Optional. Name of the tournament.
competitor_name	Array of string	Optional. List of competitors.
market_name	String	Optional. Name of the market.
outcome_name	String	Optional. Name of the outcome.

Parameter	Type	Description
scheduled	Integer	Optional. The event start time.
odds	String	Optional. The bet odds.

NB. Optional parameters can be activated in the backoffice.

NB. When a player makes bet using freebet, his balance may be either changed or not dependent on freebet type:

1. Standart Freebet - this means that only the amount of money player won will be transferred to his balance. For example, placing freebet \$5 with odd 3.5 player will get  $5 * 3.5 - 5 = 12.50$
2. Free money bet - player will receive the amount of money won and the freebet. For example, placing freebet \$5 with odd 3.5 player will get  $5 * 3.5 = 17.50$
3. No risk bet - when player make bet, the balance should be reduced by sum of freebet. In case the bet is lost, the sum of bet will be transferred to his balance. In case the bet is won, the payout will contain both the bet and won.

#### List of transaction types supported by Bet.Tech

Operation	Description
bet	Transaction for a new bet.
win	Transaction for a won bet.
refund	Transaction for a refund.
cancel	Transaction for a cancelled bet.
rollback	Transaction for a rollback.

#### List of transaction statuses supported by Bet.Tech

Code	Status	Description
0	Unconfirmed	This means the transaction is created by Bet.Tech but is not processed by Partner yet.
11	Op_accepted	This means Partner accepted the transaction and put a given amount of money on player's account on hold.
12	Op_rejected	This means Partner rejects the transaction because of internal reasons (e.g. not enough money on player's account)



Code	Status	Description
21	RM_accepted	This means the Bet.Tech risk management system accepts the bet made by player.
22	RM_rejected	This means the Bet.Tech risk management system rejects the bet made by the player. In this case a new transaction with status 32 (Refund) is generated signalling the necessity of returning money to player's balance.
23	RM_skipped	This means the Bet.Tech risk management is not going to check the bet.
31	Success	This means the status of the transaction successfully processed both by Partner and Bet.Tech
32	Refund	This means the player's bet has not successfully passed all levels of monitoring and needs to be returned to player.

### Response 200

```
{
  "id": "1660987745549553664",
  "ext_transaction_id": "1265023428769484821",
  "parent_transaction_id": null,
  "user_id": "1660986605374148608",
  "operation": "bet",
  "amount": 100000,
  "currency": "USD",
  "balance": 150000
}
```

Parameter	Type	Description
id	String	Unique identifier for transaction assigned by Partner
ext_transaction_id	String	Unique identifier of transaction assigned by Bet.Tech
parent_transaction_id	String	Unique identifier of transaction assigned by Partner
user_id	String	Unique identifier of the player assigned by Partner.
operation	String	Kind of transaction
amount	Integer	The amount of money of a given currency to be placed by player expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
balance	Integer	Balance of player after transaction is applied

NB. Bet.Tech can repeat request (with the same transaction parameters) that Partner already processed and sent response 200. In that case Bet.Tech waits the same response 200 again.

#### Response 400

```
{
  "code": 2001,
  "message": "Not enough money"
}
```

Parameter	Type	Description
code	Integer	Code of error
message	String	Text of error explaining what has happened

Error code	Message
1005	Player is blocked
1006	Player not found
1007	Session is expired
2001	Not enough money
2002	Invalid currency
2004	Bad request
2005	Invalid JWT token
3001	Bonus not found

## Method BET\_COMMIT (optional)

This request is being used to notify Partner that player's bet is accepted by Bet.Tech. In some cases after BET\_MAKE request Bet.Tech might reject the bet.

#### Request

POST https://URL/bet/commit

```
{
  "bet_transaction_id": "1665023074946387968"
}
```

Parameter	Type	Description
bet_transaction_id	String	"Transaction ID assigned by Partner when make_bet request is processed"

Response 200

Bet.Tech does not require any payload provided in Response 200.

**Method BET\_SETTLEMENT (optional)**

This request is used to notify Partner the bet status has changed.

Request

POST https://URL/bet/settlement

```
{
  "status": "win/lost",
  "bet_transaction_id": "1665023074946387968",
  "selections": [
    {"id": "1763238570711191552", "event_id": "1763230610287108096", "status": "won"},
    {"id": "1763238570711191553", "event_id": "1763128483497779200", "status": "lost"}
  ]
}
```

Parameter	Type	Description
bet_transaction_id	String	Transaction ID assigned by Partner when bet_make request is processed
status	String	Updated status of betslip: either "win", "lost", "refund", "cancel" or "rollback"
selections	String	Is used for combo and system bets only in case they lost. Provides statuses of all selections included (open, lost, won)

**<SelectionsItem>**

Parameter	Type	Description
id	String	Bet ID assigned by Partner when bet_make request is processed
event_id	String	Unique identifier of the event
selections	String	Status of the selection: "won", "lost", "open".

Response 200

Bet.Tech does not require any payload provided in Response 200.

## Method BET\_REFUND

This request is needed when the bet is declined after transaction is processed on Partner's side. Partner must carry out Responses as described below.

### Request

POST https://URL/bet/refund

```
{
  "bet_transaction_id": "1665023074946387968",
  "reason": "Some reason...",
  "bonus_id": "1664976298830860288",
  "transaction": {
    "id": "1265023428769484821",
    "betslip_id": "1659603658884648961",
    "player_id": "1659299365132570624",
    "operator_id": "1657013002915142201",
    "operator_brand_id": "1653815133341880320",
    "ext_player_id": "1659297400285696000",
    "timestamp": 1538654560.2354896,
    "status": 0,
    "amount": 1000,
    "currency": "USD",
    "operation": "refund",
    "bonus_id": "1664976298830860288"
  }
}
```

Parameter	Type	Description
bet_transaction_id	String	Transaction ID assigned by Partner to be cancelled
reason	String	Reason why transaction is going to be processed under refund procedure
bonus_id	String	Optional parameter. Bet.Tech Bonus ID
transaction	<TransactionItem>	Dictionary containing description of transaction.
<b>Refund reason status</b>		
Rejected by risk management system		
Refund bonus money		
Manual bet cancel		

### <TransactionItem>

Parameter	Type	Description
id	String	Unique identifier of transaction assigned by Bet.Tech
betslip_id	String	Unique identifier of betslip assigned by Bet.Tech when bet is made.
player_id	String	Unique identifier of player assigned by Bet.Tech.
operator_id	String	Partner unique identifier.
operator_brand_id	String	Partner Website unique identifier.
ext_player_id	String	Unique identifier assigned to a player on Partner side.
timestamp	Float	Transaction timestamp in unixtime.
status	Integer	Bet.Tech internal status for transaction described above.
amount	Integer	The sum of money to be returned from players balance expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
operation	String	Type of operation type (bet/win/refund)
bonus_id	String	Optional parameter. Bet.Tech Bonus ID

### Response 200

```
{
  "id": "1665023428769484800",
  "ext_transaction_id": "1265023428769484821",
  "parent_transaction_id": "1665023074946387968",
  "user_id": "1660986605374148608",
  "operation": "refund",
  "amount": 1000,
  "currency": "USD",
  "balance": 197000
}
```

Parameter	Type	Description
id	String	Unique identifier assigned by Partner to a given transaction
ext_transaction_id	String	Unique identifier assigned by Bet.Tech to a given transaction
parent_transaction_id	String	Unique identifier assigned by Partner to a parent transaction

Parameter	Type	Description
user_id	String	Unique identifier of player assigned by Partner.
operation	String	Type of operation processed
amount	Integer	The sum of money to be refunded expressed in cents
balance	Integer	Player balance

NB. Bet.Tech can repeat request (with the same transaction parameters) that Partner has already processed and responded 200. In that case Bet.Tech waits for the same response 200 again.

#### Response 400

```
{
  "code": 2003,
  "message": "Parent transaction not found "
}
```

Parameter	Type	Description
code	Integer	Code of error
message	String	Text of error explaining what has happened

Error code	Message
2003	Parent transaction not found
2004	Bad request
2005	Invalid JWT token

## Method BET\_WIN

The request is made by Bet.Tech when player's bet is won or cashed out. Partner must carry out Response as described below.

#### Request

POST https://URL/bet/win

```
{
  "amount": 1000,
  "currency": "USD",
}
```

```

"is_cashout": false,
"bet_transaction_id": 1659577478391402496,
"transaction":
{
  "id": "1265023428769484821",
  "betslip_id": "1659603658884648961",
  "player_id": "1659299365132570624",
  "operator_id": "1657013002915142201",
  "operator_brand_id": "1653815133341880320",
  "ext_player_id": "1659297400285696000",
  "timestamp": 1538654560.2354896,
  "status": 0,
  "amount": 1000,
  "currency": "USD",
  "operation": "win",
  "balance": 0,
  "bonus_id": "1664976298830860288"
}
}

```

Parameter	Type	Description
amount	Integer	Amount of money to be transferred to players balance.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
is_cashout	Boolean	The mark of cash out being applied.
bet_transaction_id	String	Unique identifier assigned by Partner that cash out to be applied on
transaction	<TransactionItem>	Dictionary containing description of transaction.

#### <TransactionItem>

Parameter	Type	Description
id	String	Unique identifier of transaction assigned by Bet.Tech
betslip_id	String	Unique identifier of betslip assigned by Bet.Tech when bet is made.
player_id	String	Unique identifier of player assigned by Bet.Tech.
operator_id	String	Partner unique identifier assigned by Bet.Tech
operator_brand_id	String	Partner Website unique identifier assigned by Bet.Tech
ext_player_id	String	Unique identifier assigned to a player on Partner side.
timestamp	Float	Transaction timestamp in unixtime.

Parameter	Type	Description
status	Integer	Bet.Tech internal status for transaction described above.
amount	Integer	The sum of money to be returned from players balance expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
operation	String	Type of operation type (bet/win/refund)
bonus_id	String	Optional parameter. Bet.Tech Bonus ID

NB. Player is able to request a cash out when Bet.Tech meets the conditions of bet. In this case Bet.Tech creates the new transaction request with the flag 'is\_cashout' = True

#### Response 200

```
{
  "id": "1665023428769484800",
  "ext_transaction_id": "1265023428769484821",
  "parent_transaction_id": "1665023074946387968",
  "user_id": "1660986605374148608",
  "operation": "win",
  "amount": 1000,
  "currency": "USD",
  "balance": 197000
}
```

Parameter	Type	Description
id	String	Unique identifier assigned by Partner to a given transaction
ext_transaction_id	String	Unique identifier assigned by Bet.Tech to a given transaction
parent_transaction_id	String	Unique identifier of parent transaction assigned by Partner
user_id	String	Unique identifier of player assigned by Partner.
operation	String	Type of operation to be performed
amount	Integer	The sum of money to be refunded expressed in cents
balance	Integer	Player balance



NB. Bet.Tech can repeat request (with the same transaction parameters) that Partner has already processed and responded 200. In that case Bet.Tech waits for the same response 200 again.

#### Response 400

```
{
  "code": 2003,
  "message": "Parent transaction not found"
}
```

Parameter	Type	Description
code	Integer	Code of error
message	String	Text of error explaining what has happened

Error code	Message
2003	Parent transaction not found
2004	Bad request
2005	Invalid JWT token

## Method BET\_CANCEL

The request is used in case when player's bet was cancelled manually. Method BET\_CANCEL is used only for bets with status "won".

Previous bet status	New bet status	Action
Won	Cancelled	Refund bet stake to player's balance (bet_refund), take win amount out of the balance (bet_cancel). Be aware of the situation when the funds, which have been previously credited to the player, are no longer available.
Open	Cancelled	Refund bet stake to player's balance (bet_refund)
Lost	Cancelled	Refund bet stake to player's balance (bet_refund)

#### Request

POST https://URL/bet/cancel

```
{
  "bet_transaction_id": "1665023074946387968",
  "reason": "Back office triggered bet cancellation",
  "transaction": {
    "id": "1265023428769484821",
    "betslip_id": "1659603658884648961",
    "player_id": "1659299365132570624",
    "operator_id": "1657013002915142201",
    "operator_brand_id": "1653815133341880320",
    "ext_player_id": "1659297400285696000",
    "timestamp": 1538654560.2354896,
    "status": 0,
    "amount": 1000,
    "currency": "USD",
    "operation": "cancel",
    "balance": 0,
    "bonus_id": "1664976298830860288"
  }
}
```

Parameter	Type	Description
bet_transaction_id	String	Unique identifier assigned by Partner that cash out to be applied on
win_transaction_id	String	Transaction identifier assigned by Partner, received by /bet/win call (if it has been appeared)
reason	String	Reason for performing bet cancel
transaction	<TransactionItem>	Dictionary containing description of transaction.

#### <TransactionItem>

Parameter	Type	Description
id	String	Unique identifier of transaction assigned by Bet.Tech
betslip_id	String	Unique identifier of betslip assigned by Bet.Tech when bet is made.
player_id	String	Unique identifier of player assigned by Bet.Tech.
operator_id	String	Partner unique identifier assigned by Bet.Tech
operator_brand_id	String	Partner Website unique identifier assigned by Bet.Tech
ext_player_id	String	Unique identifier assigned to a player on Partner side.
timestamp	Float	Transaction timestamp in unixtime.

Parameter	Type	Description
status	Integer	Bet.Tech internal status for transaction described above.
amount	Integer	The sum of money to be returned from players balance expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
operation	String	Type of operation type (bet/win/refund)
bonus_id	String	Optional parameter. Bet.Tech Bonus ID

Response 200

```
{
  "id": "1665023428769484800",
  "ext_transaction_id": "1265023428769484821",
  "parent_transaction_id": "1665023074946387968",
  "user_id": "1660986605374148608",
  "operation": "cancel",
  "amount": 1000,
  "currency": "USD",
  "balance": 197000
}
```

Parameter	Type	Description
id	String	Unique identifier assigned by Partner to a given transaction
ext_transaction_id	String	Unique identifier assigned by Bet.Tech to a given transaction
parent_transaction_id	String	Unique identifier of parent transaction assigned by Partner
user_id	String	Unique identifier of player assigned by Partner.
operation	String	Type of operation to be performed
amount	Integer	The sum of money to be refunded expressed in cents
balance	Integer	Player balance

Response 400

```
{
  "code": 2003,
```

```
"message": "Parent transaction not found"
}
```

Parameter	Type	Description
code	Integer	Code of error
message	String	Text of error explaining what has happened

Error code	Message
2003	Parent transaction not found
2004	Bad request
2005	Invalid JWT token

## Method BET\_DISCARD

This method is triggered in case of fail during bet\_make procedure. Currently there could be two main reasons resulting in this method: connection lost or unsatisfactory result.

### Request

POST https://URL/bet/discard

```
{
  "transaction_id": "123456789",
  "reason": "Lost connection"
}
```

Parameter	Type	Description
transaction_id	String	
reason	String	Descriptions of why this method being called

### Response 200

Bet.Tech does not require any payload provided in Response 200.

## Method BET\_ROLLBACK

The request is used in case Bet.tech made a bet's rollback. Rollback happens when the bet is originally settled one way, but later (can be minutes or sometimes even days after) the results are changed and the bet gets settled differently. A bet's rollback

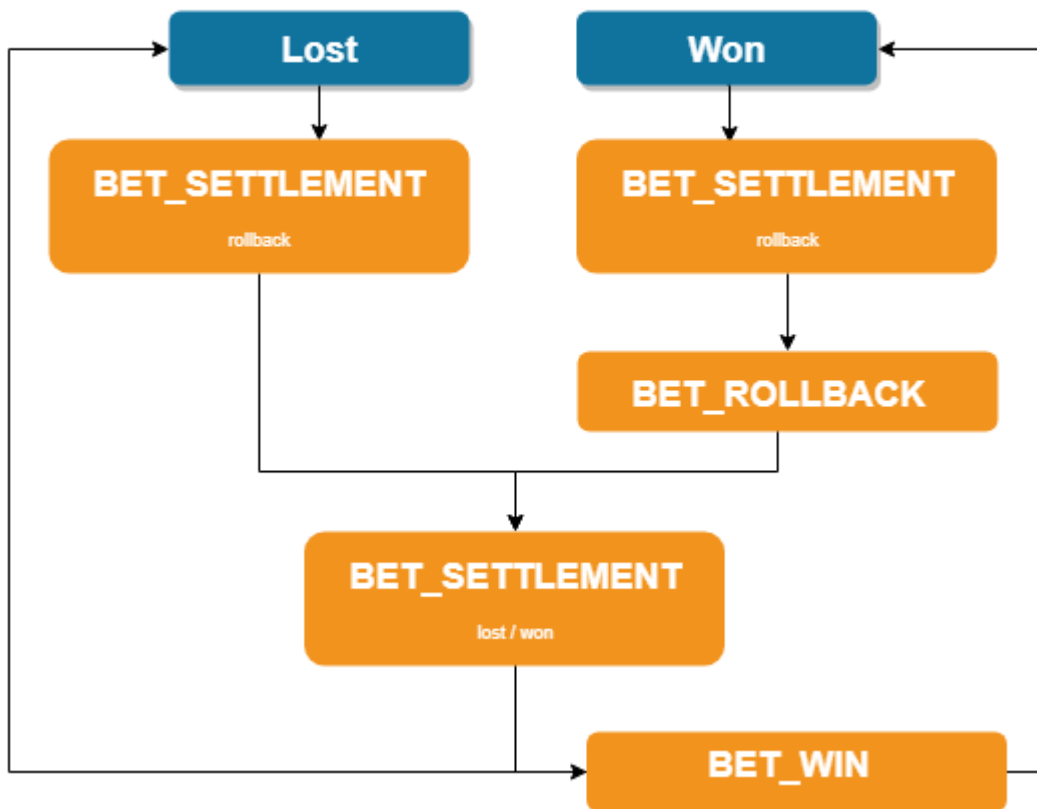
means that bet status changes to the previous bet status (open) and then it can be settled to the opposite. It is possible to make rollback only in case the bet is settled. The status is changed by method BET\_SETTLEMENT. In case the bet was won we call BET\_ROLLBACKS method to rollback win amount out of the player's balance. Be aware of the situation when the funds, which have been previously credited to the player, are no longer available. There are a few ways how to operate in this case.

1. The player's balance gets negative. Here is a chance that a player won't deposit funds to a negative account.
2. The balance displays 0 but actually is negative. Here is a chance that a player will complain about the deposit amount and balance difference.
3. The rollback call happens at regular intervals til balance will be deposited and the request processed. There is no need to support a negative balance.

Previous bet status	New bet status	Action
Won	Rollback	Rollback win amount out of the balance. Be aware of the situation when the funds, which have been previously credited to the player, are no longer available.
Lost	Rollback	No action.
Rollback	Lost	No action.
Rollback	Won	Add win amount to a player's balance.

## A bet's cycle of life

### ROLLBACK SCHEME



#### Request

POST https://URL/bet/rollback

```

{
  "bet_transaction_id": "1665023074946387968",
  "win_transaction_id": "2307166563874946502",
  "transaction":
  {
    "id": "1265023428769484821",
    "betslip_id": "1659603658884648961",
    "player_id": "1659299365132570624",
    "operator_id": "1657013002915142201",
    "operator_brand_id": "1653815133341880320",
    "ext_player_id": "1659297400285696000",
    "timestamp": 1538654560.2354896,
    "status": 0,
    "amount": 1000,
    "currency": "USD",
    "operation": "rollback",
    "balance": 0,
    "bonus_id": "1664976298830860288"
  }
}

```

Parameter	Type	Description
bet_transaction_id	String	Transaction identifier assigned by Partner, received by /bet_make call
win_transaction_id	String	Transaction identifier assigned by Partner, received by /bet_win call
transaction	<TransactionItem>	Dictionary containing information about the transaction.

**<TransactionItem>**

Parameter	Type	Description
id	String	Unique identifier of transaction assigned by Bet.Tech
betslip_id	String	Unique identifier of betslip assigned by Bet.Tech when a bet is made.
player_id	String	Unique identifier of a player assigned by Bet.Tech.
operator_id	String	Partner unique identifier assigned by Bet.Tech
operator_brand_id	String	Partner Website unique identifier assigned by Bet.Tech
ext_player_id	String	Unique identifier assigned to a player on Partner side.
timestamp	Float	Transaction timestamp in unixtime.
status	Integer	Bet.Tech internal status for transaction described above.
amount	Integer	The sum of money to be returned from players balance expressed in cents.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
operation	String	Type of operation type (bet/win/refund)
bonus_id	String	Optional parameter. Bet.Tech Bonus ID

Response 200

```
{
  "id": "1665023428769484800",
  "ext_transaction_id": "1265023428769484821",
  "parent_transaction_id": "1665023074946387968",
  "user_id": "1660986605374148608",
  "operation": "rollback",
  "amount": 1000,
  "currency": "USD",
```

```
"balance": 197000
}
```

Parameter	Type	Description
id	String	A unique identifier assigned by Partner to a given transaction
ext_transaction_id	String	A unique identifier assigned by Bet.Tech to a given transaction
parent_transaction_id	String	Unique identifier of parent transaction assigned by Partner
user_id	String	Unique identifier of a player assigned by Partner.
operation	String	Type of operation to be performed
amount	Integer	The sum of money to be refunded expressed in cents
balance	Integer	Player balance

#### Response 400

```
{
  "code": 2003,
  "message": "Parent transaction not found"
}
```

Parameter	Type	Description
code	Integer	Code of error
message	String	Text of error explaining what has happened
Error code		Message
2003		Parent transaction not found
2004		Bad request
2005		Invalid JWT token

## External API

This section describes External API interface. The current version of the document focuses mostly on the methods connected to the Bonus management process.

NB. Content-type for payloads is 'application/json'.



## Method PING

This method is used to identify whether Bet.Tech External API is running correctly.

### Request

GET https://BETTECH\_EXTERNAL\_API\_URL/api/v1/external\_api/ping

### Response 200

In case API is running correctly, Partner will get the following response:

```
{  
  "status": "ok"  
}
```

Parameter	Type	Description
Status	String	The value 'ok' stands for everything is running correctly

## Bonus API

This section describes API interface provided by Bet.Tech for Partner to support bonus management processes.

Bet.Tech allows Partners to create freebet templates using Backoffice. After freebet template is created, Partner can interact with it via API issuing bonuses to players, revoking them etc.

## Method TEMPLATES

This method is used for getting the list of bonus templates available for Partner to handle with.

### Request

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus/templates -d '{"operator_id":  
1657013002915142201}'
```

Parameter	Description
operator_id	Unique Partner Identifier assigned by Bet.Tech

### Response 200

```
{  
  "items":
```

```
[
  {
    "id": "1684188993169788928",
    "name": "test_direct_tennis_live",
    "is_active": false,
    "max_bonus_number": 999999,
    "type": "freebet",
    "operator_id": "1657013002915142201",
    "event_scheduled": 1646946993,
    "brand_id": "1653815133341880320",
    "from_time": 1544436660,
    "to_time": 1544868660,
    "days_to_use": 3,
    "events_availability": true,
    "restrictions":
    {
      "restriction_events":
      [
        {
          "sport_id": null,
          "sport_type": null,
          "category_id": null,
          "tournament_id": null,
          "event_id": "1683937560088416256"
        }
      ],
      "type": "all"
    },
    "freebet_data":
    {
      "is_api_amount": true,
      "amount_list": [
        {
          "amount": 1000,
          "max_cap": 100000,
          "currency": "USD"
        }
      ],
      "type": "bet_refund",
      "min_selection": 1,
      "max_selection": 1,
      "min_odd": 1.01,
      "max_odd": 1000
    },
    "descriptions":
    {
      "en": "Tennis"
    }
  }
]
```

Parameter	Type	Description
items	List of <TemplateItem>	Array of dictionaries containing information about templates.

## &lt;TemplateItem&gt;

Parameter	Type	Description
id	String	Unique bonus template identifier assigned by Bet.Tech
name	String	The name of the bonus template assigned by Partner in Backoffice
is_active	Boolean	The mark of whether the template is ready to be used when issuing bonuses to players or not.
max_bonus_number	Integer	The amount of bonuses can be issued per player following the given bonus template
type	String	The type of the bonus template (either freebet or comboboot)
operator_id	String	Partner unique identifier assigned by Bet.Tech
brand_id	String	Partner Website unique identifier assigned by Bet.Tech
event_scheduled	Integer	It is possible to make a freebet on events started before the specified time. If the freebet is allowed for Live and the event started before the specified time, it is possible to make a bet on that event until it is over. For example, Event scheduled is specified as 22:30, event starts at 22:30, then live bet is available until the event is over.
from_time	Float	Timestamp in Unixtime showing the date from which the template can be used to issue bonuses.
to_time	Float	Timestamp in Unixtime showing the date until which the template can be used to issue bonuses. After this date request to generate bonuses will cause errors.
days_to_use	Integer	The number of days player has to use bonus after activation.
events_availability	Boolean	Not used

Parameter	Type	Description
restrictions	<RestrictionsList>	Dictionary containing restrictions for using bonuses.
freebet_data	<FreebetDataItem>	The dictionary describing terms of generating bonuses under a given template.
comboboost_data	<ComboboostDataItem>	The dictionary describing terms of generating bonuses under a given template.
descriptions	Dictionary	The dictionary contains pair-wised information in format of "language: description" allowing Partner to localize the description of the bonus that can be seen by player.

NB. There can be only one field with bonus data: freebet\_data or comboboost\_data.

#### <RestrictionsList>

Parameter	Type	Description
restriction_events	List of <RestrictionItem>	The array of dictionaries containing information about restrictions.
type	String	The mark of what kind of restriction can be applied to the bonus - "all", "live", "prematch".

#### <RestrictionItem>

Parameter	Type	Description
sport_id	Integer	Unique identifier of a sport the bonus can be applied to.
category_id	Integer	Unique identifier of the sport's category the bonus can be applied to.
tournament_id	Integer	Unique identifier of a tournament the bonus can be applied to.
event_id	Integer	Unique identifier of the event the bonus can be applied to.

#### <FreebetDataItem>

Parameter	Type	Description
is_api_amount	Boolean	The mark indicating the method of how freebet amount is generated when bonus assigned to a players. In case of TRUE, the Partner is required to add the amount of freebet to request. Otherwise, the amount of freebet to be given to a player must be configures using backoffice.
amount_list	List of <AmountItem>	The array of dictionaries containing information about the amount of freebet to be given connecting to different currencies.
type	String	The indicator of what kind of freebet will be used when issuing bonuses to a player under this template. The values supported: bet_refund, free_money, snr
min_selection	Integer	The minimal number of selections in betslip a player required to add to apply for a given bonus.
max_selection	Integer	The maximum number of selections in betslip a player required to add to apply for a given bonus
min_odd	Float	The minimum odd each selection of the betslip must meet to use bonus.
max_odd	Float	The maximum odd each selection of the betslip must meet to use bonus.

**<ComboboostDataItem>**

Parameter	Type	Description
min_odd	Float	Minimal odds for each bet
multipliers	List of float	List of multipliers depends on events number
total_multiplier	Float	The multiplier used for the bet
is_global	Boolean	The bonus is permanently available for all players without granting

**<AmountItem>**

Parameter	Type	Description
amount	Integer	The amount of freebet to be given to a player expressed in cents.
max_cap	Integer	The maximum possible win amount.

Parameter	Type	Description
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.

Error

```
{
  "error":
  {
    "name": "TemplateNotFoundError",
    "description": "Template 1684188993169788922 not found"
  }
}
```

Parameter	Type	Description
error	<ErrorItem>	Dictionary containing information of what has happened when requesting endpoint.

**<ErrorItem>**

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong
## Method TEMPLATE		

This method is used to get information about a particular bonus template. The successfull response will be the same as described in method Templates - <TemplateItem>.

Request

POST https://BETTECH\_EXTERNAL\_API\_URL/api/v1/external\_api/bonus/template

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus/templates -d '{"operator_id": 1657013002915142201, "template_id": 1657013872955142242}'
```

Response

See <TemplateItem> described above

Error

```
{
  "error":
  {
```

```
{
  "name": "TemplateNotFoundError",
  "description": "Template 1684188993169788922 not found"
}
```

Parameter	Type	Description
error	<ErrorItem>	Dictionary containing information of what has happened when requesting endpoint.

#### <ErrorItem>

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong

## Method PLAYER\_BONUSES

This method is used to get the list of bonuses of a particular player.

#### Request

POST https://BETTECH\_EXTERNAL\_API\_URL/api/v1/external\_api/bonus/player\_bonuses

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus/player_bonuses -d '{"brand_id": 1653815133341880320, "external_player_id": "1667142267007275008"}'
```

Parameter	Type	Description
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
external_player_id	String	Unique identifier assigned to a player on Partner side.

#### Response

```
{
  "items":
  [
    {
      "id": "1687158762034565120",
      "template_id": "1687153619004166144",
      "name": "demo_inv_snr",
      "type": "freebet",
      "player_id": "1666438116296232960",
      "external_player_id": "166643811112073216",
      "brand_id": "1653815133341880320",
      "event_scheduled": 1646946993,
      "receipt_date": 1545224208,
      "issue_type": "api",
    }
  ]
}
```

```

    "restrictions":
    {
      "restriction_events":
      [
        {
          "sport_id": null,
          "sport_type": null,
          "category_id": null,
          "tournament_id": null,
          "event_id": "1686751907374309376"
        }
      ],
      "type": "all"
    },
    "viewed": true,
    "activation_date": 1545224304.505800724,
    "ending_date": 1545310704.505800724,
    "status": "activated",
    "from_time": 1545143520,
    "to_time": 1546093920,
    "freebet_data":
    {
      "amount": 3000,
      "max_cap": 19000,
      "currency": "USD",
      "exchange_rate": "0.876899",
      "type": "snr",
      "min_selection": 1,
      "max_selection": 1,
      "min_odd": 1.01,
      "max_odd": 1000
    }
  }
]
}

```

Parameter	Type	Description
items	List of <BonusItem>	The array containing information about bonuses of the requested player

#### <BonusItem>

Parameter	Type	Description
id	Integer	Unique bonus identifier assigned by Bet.Tech.
template_id	Integer	Unique bonus template identifier assigned by Bet.Tech
name	String	The name of the bonus template assigned by Partner in Backoffice



Parameter	Type	Description
type	String	The type of the bonus template (either freebet or comboboot)
player_id	Integer	Unique player identifier assigned by Bet.Tech
external_player_id	String	Unique player identifier assigned by Partner
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
event_scheduled	Integer	It is possible to make a freebet on events started before the specified time. If the freebet is allowed for Live and the event started before the specified time, it is possible to make a bet on that event until it is over. For example, Event scheduled is specified as 22:30, event starts at 22:30, then live bet is available until the event is over.
receipt_date	Float	Timestamp in Unixtime expressing the datetime when player received bonus.
issue_type	String	The indicator of bonus issue method. In case the bonus is given through API interface it will be "api".
restrictions	<RestrictionsList>	Dictionary containing information about restrictions a player must fulfill to use a given bonus.
viewed	Boolean	The indicator of whether the bonus has been viewed by player or not.
activation_date	Float	Timestamp in Unixtime indicating the datetime when the bonus was activated by player.
ending_date	Float	Timestamp in Unixtime indication the datetime after which the activated bonus will expire.

Parameter	Type	Description
status	String	The current status of the bonus. For the newly issued bonuses the status will be "New". "Active" stands for bonuses being activated, but not used yet. "Activated" indicates that the bonus was successfully used. "Expired" shows that a player cannot use this bonus because of date range limitations. "Revoked" stands for bonuses revoked by Partner using either API or Backoffice. "Done"
from_time	Float	Timestamp in unixtime indicating datetime after which a player can activate the bonus.
to_time	Float	Timestamp in unixtime indicating datetime after which a player lose the opportunity to activate the bonus.
freebet_data	<FreebetDataItem>	The dictionary describing the terms of freebet.
comboboost_data	<ComboboostDataItem>	The dictionary describing terms of generating bonuses under a given template.

NB. There can be only one field with bonus data: freebet\_data or comboboost\_data.

#### <RestrictionsList>

Parameter	Type	Description
restriction_events	List of <RestrictionItem>	The array of dictionaries containing information about restrictions.
type	String	The mark of what kind of restriction can be applied to the bonus - "all", "live", "prematch".

#### <RestrictionItem>

Parameter	Type	Description
sport_id	Integer	Unique identifier of a sport the bonus can be applied to.

Parameter	Type	Description
category_id	Integer	Unique identifier of the sport's category the bonus can be applied to.
tournament_id	Integer	Unique identifier of a tournament the bonus can be applied to.
event_id	Integer	Unique identifier of the event the bonus can be applied to.

**<FreebetDataItem>**

Parameter	Type	Description
is_api_amount	Boolean	The mark indicating the method of how freebet amount is generated when bonus is assigned to players. In case of TRUE, Partner is required to add the amount of freebet to the request. Otherwise, the amount of freebet to be given to a player must be configured using Backoffice.
amount_list	List of <AmountItem>	The array of dictionaries containing information about the amount of freebet to be given connecting to different currencies.
type	String	The indicator of what kind of freebet will be used when issuing bonuses to a player following this template. The values supported: bet_refund, free_money, snr
min_selection	Integer	The minimal number of selections in betslip a player required to add to apply for a given bonus.
max_selection	Integer	The maximum number of selections in betslip a player required to add to apply for a given bonus
min_odd	Float	The minimum odd each selection of the betslip must comply with to use bonus.
max_odd	Float	The maximum odd each selection of the betslip must comply with to use bonus.

**<ComboboostDataItem>**

Parameter	Type	Description
min_odd	Float	Minimal odds for each bet
multipliers	List of float	List of multipliers depends on events number
total_multiplier	Float	The multiplier used for the bet

Parameter	Type	Description
is_global	Boolean	The bonus is permanently available for all players without granting

**<AmountItem>**

Parameter	Type	Description
amount	Integer	The amount of freebet to be given to a player expressed in cents.
max_cap	Integer	The maximum possible win amount.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.

Error

```
{
  "error":
  {
    "name": "TemplateNotFoundError",
    "description": "Template 1684188993169788922 not found"
  }
}
```

Parameter	Type	Description
error	<ErrorItem>	Dictionary containing information of what has happened when requesting endpoint.

**<ErrorItem>**

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong

## Method BONUS

This method is used to get a particular bonus information using bonus ID. The successful response will be the same as one generated by the PLAYER\_BONUSES method (described in <BonusItem>).

Request

POST [https://BETTECH\\_EXTERNAL\\_API\\_URL/api/v1/external\\_api/bonus](https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus)

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus -d "{\"brand_id\":  
1653815133341880320, \"bonus_id\": 1683578214196318208}"
```

Parameter	Type	Description
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
bonus_id	Integer	Unique bonus identifier assigned by Bet.Tech.

#### Response

See <BonusItem> described above.

#### Error

```
{  
  "error":  
  {  
    "name": "TemplateNotFoundError",  
    "description": "Template 1684188993169788922 not found"  
  }  
}
```

Parameter	Type	Description
error	<ErrorItem>	Dictionary containing information of what has happened when requesting endpoint.

#### <ErrorItem>

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong

## Method MASS\_GIVE\_BONUS

This method is used to issue bonuses to players listed in request using a particular template.

NB. In case the bonus template contains the flag “is\_api\_amount” set to TRUE, Partner must provide the exact amount of freebet to a player, who is going to get it, and indicate the currency code supported by Bet.Tech.

Otherwise, the amount of freebet will be defined by the template as it was configured in Backoffice.

In case of requesting the method and passing both template with the parameter "is\_api\_amount" set to FALSE and the exact amount of freebet to be given to a player, Bet.Tech will ignore this value using the template values defined by Backoffice.

NB. Because of technical features one request can contain around 1000 players. Please, split your list of players in case there are significantly more than 1000 players.

#### Request

POST https://BETTECH\_EXTERNAL\_API\_URL/api/v1/external\_api/bonus/mass\_give\_bonus

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus/mass_give_bonus -d '{"brand_id": 1653815133341880320, "template_id": 1683558177913835520, "players_data": [ {"external_player_id": "1667142255867203584", "currency": "USD", "amount": 900, "force_activated": false} ]}'
```

Parameter	Type	Description
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
template_id	Integer	Unique bonus template identifier assigned by Bet.Tech
players_data	List of <PlayerDataItem>	The array containing information about players to issue bonus to

#### <PlayerDataItem>

Parameter	Type	Description
external_player_id	String	Unique player identifier assigned by Partner
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.
amount	Integer	The amount of freebet to be given to a player expressed in cents.
force_activated	Boolean	True - The bonus will be activated at the granting moment. False - The bonus must be activated by a player in "My bets" section

#### Response

```
{
  "result":
```

```
{
  "items": [
    {
      "id": "1694384642255364096",
      "template_id": "1694368820443090944",
      "name": "template_for_doc",
      "type": "freebet",
      "player_id": null,
      "external_player_id": "1667142255867203584",
      "brand_id": "1653815133341880320",
      "event_scheduled": 1646946993,
      "receipt_date": 1546946993,
      "issue_type": "api",
      "restrictions": {
        "restriction_events": [],
        "type": "all"
      },
      "viewed": false,
      "activation_date": null,
      "ending_date": null,
      "status": "new",
      "from_time": 1546345500,
      "to_time": 1548743100,
      "freebet_data": {
        "amount": 900,
        "max_cap": 19000,
        "currency": "USD",
        "exchange_rate": "0.873833",
        "type": "bet_refund",
        "min_selection": 1,
        "max_selection": 1,
        "min_odd": 1.01,
        "max_odd": 1000
      }
    }
  ]
}
```

Parameter	Type	Description
result	List of <BonusPlayerItem>	The array containing information about bonus issue process.

#### <BonusPlayerItem>

Parameter	Type	Description
<player_id>	<BonusItem>	The dictionary with player_id as a key and the dictionary about the bonuses issued to this particular player. In case of failure while issuing bonus to a player this dictionary will contain information about the error described below in <BonusErrorItem>.

## &lt;BonusItem&gt;

Parameter	Type	Description
id	Integer	Unique bonus identifier assigned by Bet.Tech
template_id	Integer	Unique bonus template identifier assigned by Bet.Tech
name	String	The name of the bonus template assigned by Partner in Backoffice
type	String	The type of the bonus template (either freebet or comboboost)
player_id	Integer	Unique player identifier assigned by Bet.Tech
external_player_id	String	Unique player identifier assigned by Partner
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
event_scheduled	Integer	It is possible to make a freebet on events started before the specified time. If the freebet is allowed for Live and the event started before the specified time, it is possible to make a bet on that event until it is over. For example, Event scheduled is specified as 22:30, event starts at 22:30, then live bet is available until the event is over.

| receipt\_date | Float | Timestamp in Unixtime expressing the datetime when player received the bonus. | | issue\_type | String | The indicator of bonus issue method. In case the bonus is given through the API interface it will be "api". | | restrictions |

<RestrictionsList> | Dictionary containing information about the restrictions a player must fulfill to use a given bonus. | | viewed | Boolean | The indicator of whether bonus has been viewed by the player or not. | | activation\_date | Float | Timestamp in Unixtime indicating the datetime when the bonus was activated by the player. | | ending\_date | Float | Timestamp in Unixtime indicates the datetime after which the activated bonus will expire. | | status | String | The current status of the bonus.

For newly issued bonuses the status will be "New".

"Active" stands for the bonuses being activated, but not used yet.

"Activated" indicates that the bonus was successfully used.

"Expired" shows that a player cannot use this bonus because of date range limitations.

"Revoked" stands for bonuses revoked by Partner using either API or Backoffice. | |

from\_time | Float | Timestamp in unixtime indicating datetime after which a player can activate the bonus. | | to\_time | Float | Timestamp in unixtime indicating datetime after which a player lose the opportunity to activate bonus. | | freebet\_data |

<FreebetDataItem> | The dictionary describing the terms of freebet. | |

comboboost\_data | <ComboboostDataItem> | The dictionary describing terms of



generating bonuses under a given template. | | descriptions | Dictionary | The dictionary contains pair-wised information in format of "language: description" allowing Partner to localize the description of the bonus that can be seen by player. |

NB. There can be only one field with bonus data: freebet\_data or comboboot\_data.

#### <RestrictionsList>

Parameter	Type	Description
restriction_events	List of <RestrictionItem>	The array of dictionaries containing information about restrictions.
type	String	The mark of what kind of restriction can be applied to the bonus - "all", "live", "prematch".

#### <RestrictionItem>

Parameter	Type	Description
sport_id	Integer	Unique identifier of a sport the bonus can be applied to.
category_id	Integer	Unique identifier of the sport's category the bonus can be applied to.
tournament_id	Integer	Unique identifier of a tournament the bonus can be applied to.
event_id	Integer	Unique identifier of an event the bonus can be applied to.

#### <FreebetDataItem>

Parameter	Type	Description
is_api_amount	Boolean	The mark indicating the method of how freebet amount is generated when bonus assigned to a players. In case of TRUE, Partner is required to add the amount of freebet to request. Otherwise, the amount of freebet to be given to a player must be configured using Backoffice.
amount_list	List of <AmountItem>	The array of dictionaries containing information about the amount of freebet to be given connecting to different currencies.
type	String	The indicator of what kind of freebet will be used when issuing bonuses to a player following this template. The values supported: bet_refund, free_money, snr

Parameter	Type	Description
min_selection	Integer	The minimal number of selections in betslip a player required to add to apply for a given bonus.
max_selection	Integer	The maximum number of selections in betslip a player required to add to apply for a given bonus
min_odd	Float	The minimum odd each selection of the betslip must meet to use bonus.
max_odd	Float	The maximum odd each selection of the betslip must meet to use bonus.

## &lt;ComboboostDataItem&gt;

Parameter	Type	Description
min_odd	Float	Minimal odds for each bet
multipliers	List of float	List of multipliers depends on events number
total_multiplier	Float	The multiplier used for the bet
is_global	Boolean	The bonus is permanently available for all players without granting

## &lt;AmountItem&gt;

Parameter	Type	Description
amount	Integer	The amount of freebet to be given to a player expressed in cents.
max_cap	Integer	The maximum possible win amount.
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.

Error

```
{
  "result":
  [
    {
      "1667142255867203584":
      {
        "error":
        {
          "name": "TemplateNotFoundError",
          "description": "Template 1683558177913835520 not found"
        }
      }
    }
  ]
}
```

```

    }
  }
]
}

```

**<BonusErrorItem>**

Parameter	Type	Description
error	<ErrorItem>	The dictionary containing information about what has gone wrong

**<ErrorItem>**

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong

## Method REVOKE\_BONUS

This methods is used to revoke bonuses from a player. In case the bonus is successfully revoked Partner will get a revoked bonus item with the updated status in response.

NB. Only NEW and ACTIVE bonuses can be revoked. In case the bonus was used, it cannot be revoked using this method.

Request

```
curl -v -X POST https://BETTECH_EXTERNAL_API_URL/api/v1/external_api/bonus/revoke_bonus -d "{\"brand_id\": 1653815133341880320, \"bonuses_ids\": [1683569582905888768]}\""
```

Parameter	Type	Description
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
bonuses_ids	List of integer	The array containing bonus ids to be revoked.

Response

```

{
  "result":
  [
    {
      "1667142255867203584":
      {

```

```

    "id": "1694384642255364096",
    "template_id": "1694368820443090944",
    "name": "template_for_doc",
    "type": "freebet",
    "player_id": null,
    "event_scheduled": 1646946993,
    "external_player_id": "1667142255867203584",
    "brand_id": "1653815133341880320",
    "receipt_date": 1546946993,
    "issue_type": "api",
    "restrictions":
    {
      "restriction_events": [],
      "type": "all"
    },
    "viewed": false,
    "activation_date": null,
    "ending_date": null,
    "status": "new",
    "from_time": 1546345500,
    "to_time": 1548743100,
    "freebet_data":
    {
      "amount": 900,
      "max_cap": 19000,
      "currency": "USD",
      "exchange_rate": "0.873833",
      "type": "bet_refund",
      "min_selection": 1,
      "max_selection": 1,
      "min_odd": 1.01,
      "max_odd": 1000
    }
  }
}
]
}

```

Parameter	Type	Description
result	List of <BonusPlayerItem>	The array containing information about bonuses revoked

#### <BonusPlayerItem>

Parameter	Type	Description
<player_id>	<BonusItem>	"The dictionary with bonus_id as a key and the dictionary about the bonus revoked. In case of failure while issuing a bonus to a player this dictionary will contain information about the error described below in <BonusErrorItem>."

#### <BonusItem>

Parameter	Type	Description
id	Integer	Unique bonus identifier assigned by Bet.Tech.
template_id	Integer	Unique bonus template identifier assigned by Bet.Tech
name	String	The name of bonus template assigned by Partner in Backoffice
type	String	The type of bonus template (either freebet or comboboost)
player_id	Integer	Unique player identifier assigned by Bet.Tech
external_player_id	String	Unique player identifier assigned by Partner
brand_id	Integer	Partner Website unique identifier assigned by Bet.Tech
event_scheduled	Integer	It is possible to make a freebet on events started before the specified time. If the freebet is allowed for Live and the event started before the specified time, it is possible to make a bet on that event until it is over. For example, Event scheduled is specified as 22:30, event starts at 22:30, then live bet is available until the event is over.
receipt_date	Float	Timestamp in Unixtime expressing the datetime when player received bonus.
issue_type	String	The indicator of bonus issue method. In case the bonus is given through the API interface it will be "api".
restrictions	<RestrictionsList>	Dictionary containing information about restrictions a player must fall within to use a given bonus.
viewed	Boolean	The indicator of whether bonus has been viewed by player or not.
activation_date	Float	Timestamp in Unixtime indicating the datetime when the bonus was activated by player.

Parameter	Type	Description
ending_date	Float	Timestamp in Unixtime indicating the datetime after which the activated bonus will expire.
status	String	The current status of the bonus. For the newly issued bonuses the status will be "New". "Active" stands for the bonuses being activated but not used yet. "Activated" indicated that the bonus was successfully used. "Expired" shows that a player cannot use this bonus because of date range limitations. "Revoked" stands for bonuses revoked by Partner using either API or Backoffice.
from_time	Float	Timestamp in unixtime indicating datetime after which a player can activate the bonus.
to_time	Float	Timestamp in unixtime indicating datetime after which a player lose the opportunity to activate bonus.
freebet_data	<FreebetDataItem>	The dictionary describing the terms of freebet.
comboboost_data	<ComboboostDataItem>	The dictionary describing terms of generating bonuses under a given template.

NB. There can be only one field with bonus data: freebet\_data or comboboost\_data.

#### <RestrictionsList>

Parameter	Type	Description
restriction_events	List of <RestrictionItem>	The array of dictionaries containing information about restrictions.
type	String	The mark of what kind of restriction can be applied to the bonus - "all", "live", "prematch".

#### <RestrictionItem>

Parameter	Type	Description
sport_id	Integer	Unique identifier of a sport the bonus can be applied to.
category_id	Integer	Unique identifier of the sport's category the bonus can be applied to.
tournament_id	Integer	Unique identifier of a tournament the bonus can be applied to.
event_id	Integer	Unique identifier of the event the bonus can be applied to.

**<FreebetDataItem>**

Parameter	Type	Description
is_api_amount	Boolean	The mark indicating the method of how freebet amount is generated when bonus assigned to a players. In case of TRUE, Partner is required to add the amount of freebet to request. Otherwise, the amount of freebet to be given to a player must be configured using Backoffice.
amount_list	List of <AmountItem>	The array of dictionaries containing information about the amount of freebet to be given connecting to different currencies.
type	String	The indicator of what kind of freebet will be used when issuing bonuses to a player following this template. The values supported: bet_refund, free_money, snr
min_selection	Integer	The minimal number of selections in betslip a player required to add to apply for a given bonus.
max_selection	Integer	The maximum number of selections in betslip a player required to add to apply for a given bonus
min_odd	Float	The minimum odd each selection of the betslip must meet to use bonus.
max_odd	Float	The maximum odd each selection of the betslip must meet to use bonus.

**<AmountItem>**

Parameter	Type	Description
amount	Integer	The amount of freebet to be given to a player expressed in cents.
max_cap	Integer	The maximum possible win amount.

Parameter	Type	Description
currency	String	Currency code. The full list of currencies supported can be found in the Appendices.

### Error

```
{
  "result":
  [
    {
      "1667142255867203584":
      {
        "error":
        {
          "name": "TemplateNotFoundError",
          "description": "Template 1683558177913835520 not found"
        }
      }
    }
  ]
}
```

### <BonusErrorItem>

Parameter	Type	Description
error	<ErrorItem>	The dictionary containing information about what has gone wrong

### <ErrorItem>

Parameter	Type	Description
name	String	The name of the error
description	String	The description of what has gone wrong

## Appendices

### JWT Token example

Find packages to work with JWT here: <https://jwt.io/>

```
import time
import random

import jwt

from Crypto.PublicKey import RSA
```



```
now = int(time.time())

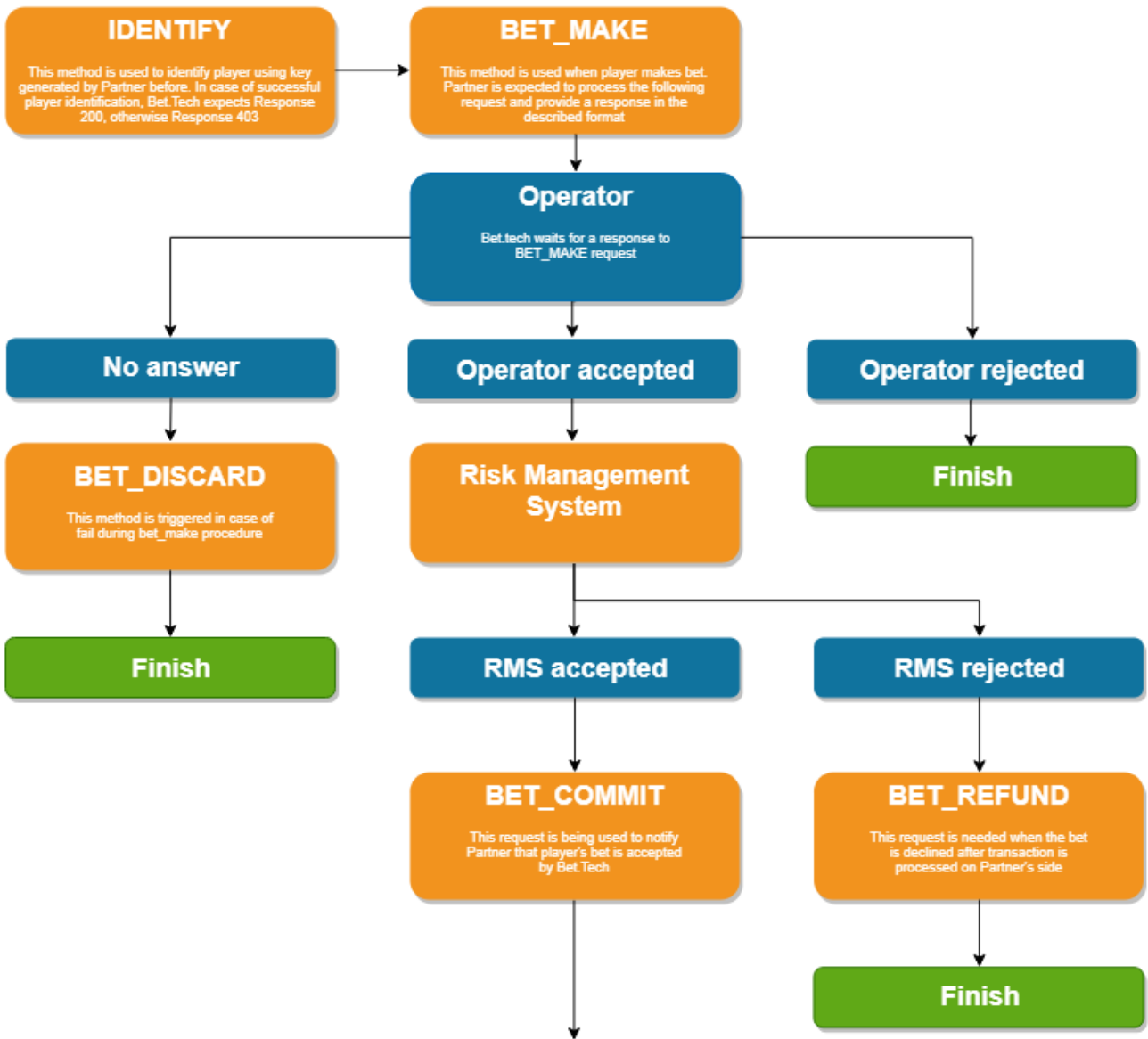
message = {
    'iat': now,
    'exp': now + 3600,
    'jti': str(random.random()),
    'iss': 'operator_id',
    'aud': 'brand_id',
    'payload': {'test': 'passed'},
    'nbf': int(time.time())
}
key = RSA.generate(4096)
private_key = key.exportKey('PEM').decode('utf-8')
public_key = key.publickey().exportKey('PEM').decode('utf-8')
token = jwt.encode(payload=message,
                    key=private_key,
                    algorithm='RS256').decode('utf-8')
print(token)

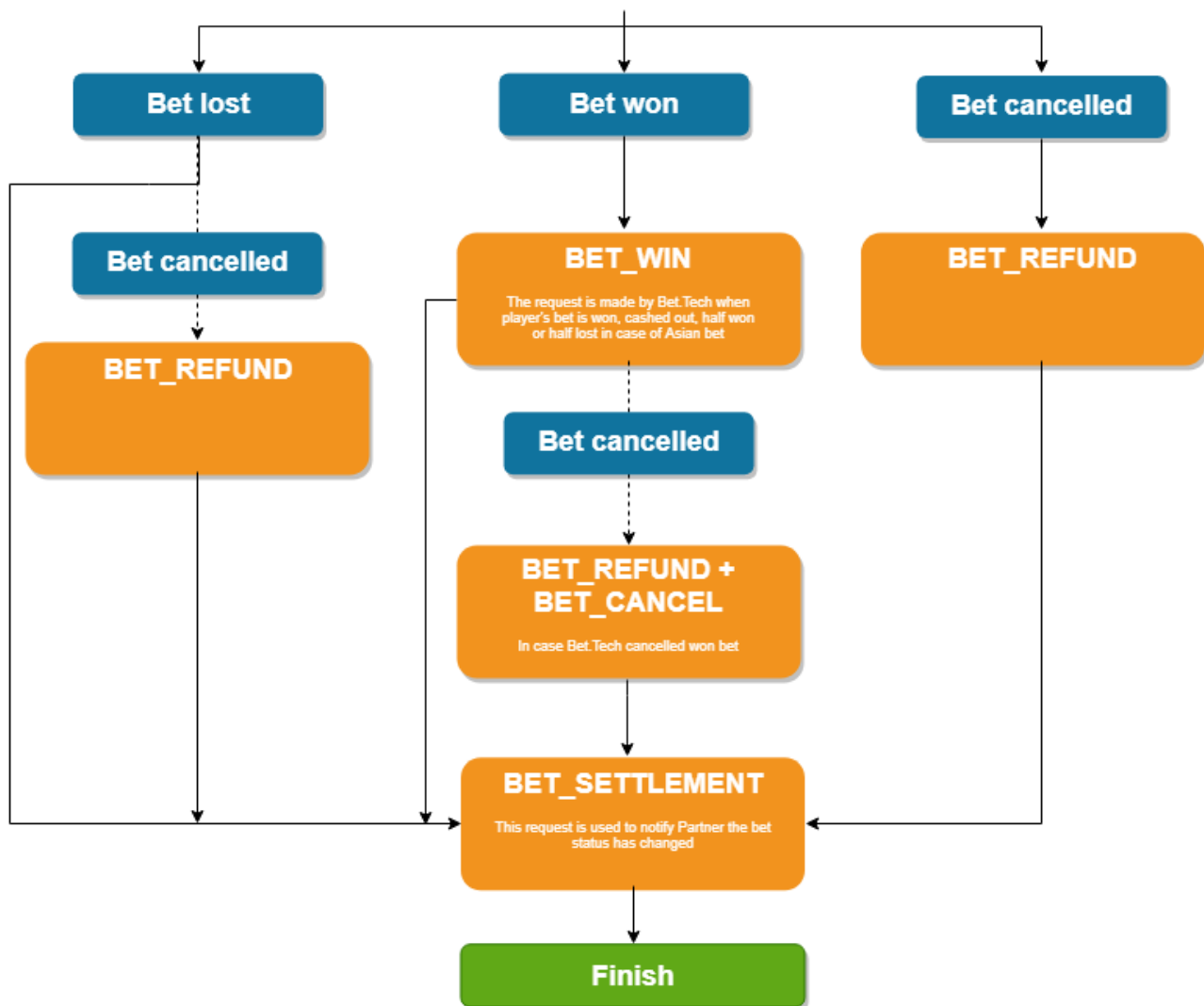
print(private_key)
print(public_key)

message = jwt.decode(jwt=token,
                     key=public_key,
                     verify=True,
                     audience='brand_id',
                     issuer='operator_id',
                     algorithms=['RS256'])

print(message)
print('PAYLOAD:', message.get('payload'))
```

## A bet's cycle of life





## Cash out

Cash out is an individual offer initiated by Sportsbook platform, addressed to a bet participant, aimed at changing one or several essential betting conditions (coefficient, event calculation time, etc.) in order to fix a new result and complete the bet at the current time (further - Cash out). The offer to redeem a bet can be both accepted and rejected by the participant of the bet. By selecting Cash out the participant of the bet confirms his acceptance of the new essential conditions of the bet. Cash out rates can be offered for both prematch and for live betting. The bookmaker's office reserves the right to change the offer to repurchase the bid over time, or not to form a bid to repurchase the bet without giving a reason.

## Types of bets

### Selection

Selection is a possible separate outcome of the event.

## Single

Single is a bet on 1 selection in one event. The abbreviation is "1/1"

## Combo

Combo bet, also known as accumulator or multibet, is a bet that combines 2 or more independent selections into a single bet. Combo bet can only be won if all selections are won. The possible winnings will be equal to the result of multiplying the odds of all selections included in this bet. The maximum selections could be 20 in one combo bet. The abbreviation is "N/N", for example, "2/2", "5/5", "12/12".

## System

System bet is a set of several individual combo bets which is a combinations of combos of the same size from a fixed set of selections. As the system bet is a set of combo bets it is possible to get return even if not all predictions are correct. A minimum one combo must be correct to get a return. Please note that the amount shown under the potential win is the maximum amount can be won if all predictions are correct. Furthermore, the stake is placed for one combo will be multiplied according to the number of combo bets created through the system bet. Three different selections is the minimum number for a system bet, 20 is the maximum. A System N from M means that there are chosen M selections and every combo contains N of them. For example, you have chosen four different selections (A, B, C, D) and opted for a 2/4 system bet. This means that you will be placing a total of six combo bets, since there are six possible winning pairs based on those four selections (AB, AC, AD, BC, BD, CD). The abbreviation is "N/M", for example, "2/3", "5/8".

## Trixie

Trixie is a kind of system bet (full-covered). A trixie consists of 4 bets involving 3 selections in different events. The bet includes 3 doubles and 1 treble. A minimum of 2 of your selections must be successful to get a return. For example, if your 3 selections were A, B and C, then your 4 bets would be AB, AC, BC, and ABC. The abbreviation is "F/3".

## Patent

Patent is a kind of system bet (full-covered with singles). A Patent consists of 7 bets involving 3 selections in different events. The bet includes a single on each selection, plus 3 doubles and 1 treble. Just one successful selection guarantees a return. For example, if your 3 selections were A, B, and C, then your 7 bets would be A, B, C, AB, AC, BC, and ABC. The abbreviation is "FS/3".

## Yankee

Yankee is a kind of system bet. A Yankee consists of 11 bets involving 4 selections in different events. The bet includes 6 doubles, 4 trebles, and a combo. A minimum of 2 of

your selections must be successful to get a return. For example, if your 4 selections were A, B, C, and D, then your 11 bets would be AB, AC, AC, BC, BD, CD, ABC, ABD, BCD, ACD and a Four-Fold combo ABCD. The abbreviation is "F/4".

### **Lucky 15**

Lucky 15 is a kind of system bet. A Lucky 15 consists of 15 bets involving 4 selections in different events. The bet includes 4 singles, 6 doubles, 4 trebles, and 1 four-fold. Just one successful selection guarantees a return. For example, if your 4 selections were A, B, C, and D, then your 15 bets would be A, B, C, D, AB, AC, AC, BC, BD, CD, ABC, ABD, BCD, ACD and a Four-Fold combo ABCD. The abbreviation is "FS/4".

### **Canadian**

Canadian is a kind of system bet. A Canadian (also known as a Super Yankee) consists of 26 bets involving 5 selections in different events. The bet includes 10 doubles, 10 trebles, 5 four-folds plus a combo. A minimum of 2 of your selections must be successful to get a return. The abbreviation is "F/5".

### **Lucky 31**

Lucky 31 is a kind of system bet. A Lucky 31 consists of 31 bets involving 5 selections in different events. The bets include 5 singles, 10 doubles, 10 trebles, 5 four-folds and a five-fold combo. Just one successful selection guarantees a return. The abbreviation is "FS/5".

### **Heinz**

Heinz is a kind of system bonus. A Heinz consists of 57 bets involving 6 selections in different events. The bet includes 15 doubles, 20 trebles, 15 four-folds, 6 five-folds and a six-fold combo. A minimum of 2 of your selections must be successful to get a return. The abbreviation is "F/6".

### **Lucky 63**

Lucky 63 is a kind of system bonus. A Lucky 63 consists of 63 bets involving 6 selections in different events. The bet includes 6 singles, 15 doubles, 20 trebles, 15 four-folds, 6 five-folds and a six-fold combo. Just one successful selection guarantees a return. The abbreviation is "FS/6".

### **Super Heinz**

Super Hienz is a kind of system bonus. A Super Heinz consists of 120 bets involving 7 selections in different events. The bet includes 21 doubles, 35 trebles, 35 four-folds, 21 five-folds, 7 six-folds and a seven-fold combo. A minimum of 2 of your selections must be successful to get a return. The abbreviation is "F/7".

### **Goliath**

Goliath is a kind of system bonus. A Goliath consists of 247 bets involving 8 selections in

different events. The bet includes 28 doubles, 56 trebles, 70 four-folds, 56 five-folds, 28 six-folds, 8 seven-folds and an eight-fold combo. A minimum of 2 of your selections must be successful to get a return. The abbreviation is "F/8".

## Language Support

Please make sure to contact your Bet.Tech Account Manager to enable list of languages supported by your Platform.

Abbreviation	Description
ar	Arabic
az	Azerbaijan
bg	Bulgarian
bs	Bosnian
cs	Czech
da	Danish
de	German
el	Greek
en	English
es	Spanish
et	Estonian
fi	Finnish
fr	French
he	Hebrew
hr	Croatian
hu	Hungarian
id	Indonesian
it	Italian
ja	Japanese
ka	Georgian
kk	Kazakh
ko	Korean

Abbreviation	Description
lt	Lithuanian
lv	Latvian
mk	Macedonian
nl	Dutsch
no	Norwegian
pl	Polish
pt	Portuguese
pt-BR	Brazilian Portuguese
ro	Romanian
ru	Russian
sk	Slovak
sl	Slovenian
sq	Albanian
sr	Serbian
sr-Latn	Serbian Latin
sv	Swedish
th	Thai
tr	Turkish
uk	Ukranian
vi	Vietnamese
zh	Chinese (simplified), also zh-Hans
zh-Hant	Chinese (traditional)

## Currency Support

Please make sure to contact your Bet.Tech Account Manager to enable list of currencies supported by your Platform.

Abbreviation	Description
AED	UAE Dirham

Abbreviation	Description
AFN	Afghan Afghani
ALL	Albanian Lek
AMD	Armenian Dram
ANG	Netherlands Antillean Gulden
AOA	Angolan Kwanza
ARS	Argentine Peso
AUD	Australian Dollar
AWG	Aruban Florin
AZN	Azerbaijani Manat
BAM	Bosnia And Herzegovina Konvertibilna Marka
BBD	Barbadian Dollar
BDT	Bangladeshi Taka
BGN	Bulgarian Lev
BHD	Bahraini Dinar
BIF	Burundi Franc
BND	Brunei Dollar
BOB	Bolivian Boliviano
BRL	Brazilian Real
BSD	Bahamian Dollar
BTC	Bitcoin
BTN	Bhutanese Ngultrum
BWP	Botswana Pula
BYN	New Belarusian Ruble
BYR	Belarusian Ruble
BZD	Belize Dollar
CAD	Canadian Dollar
CDF	Congolese Franc



Abbreviation	Description
CHF	Swiss Franc
CLP	Chilean Peso
CNY	Chinese Yuan
COP	Colombian Peso
CRC	Costa Rican Colon
CUP	Cuban Peso
CVE	Cape Verdean Escudo
CZK	Czech Koruna
DJF	Djiboutian Franc
DKK	Danish Krone
DOP	Dominican Peso
DZD	Algerian Dinar
EGP	Egyptian Pound
ERN	Eritrean Nakfa
ETB	Ethiopian Birr
EUR	Euro
FJD	Fijian Dollar
FKP	Falkland Islands Pound
GBP	British Pound
GEL	Georgian Lari
GHS	Ghanaian Cedi
GIP	Gibraltar Pound
GMD	Gambian Dalasi
GNF	Guinean Franc
GTQ	Guatemalan Quetzal
GYD	Guyanese Dollar
HKD	Hong Kong Dollar

Abbreviation	Description
HNL	Honduran Lempira
HRK	Croatian Kuna
HTG	Haitian Gourde
HUF	Hungarian Forint
IDR	Indonesian Rupiah
ILS	Israeli New Sheqel
INR	Indian Rupee
IQD	Iraqi Dinar
IRR	Iranian Rial
ISK	Icelandic Króna
JMD	Jamaican Dollar
JOD	Jordanian Dinar
JPY	Japanese Yen
KES	Kenyan Shilling
KGS	Kyrgyzstani Som
KHR	Cambodian Riel
KMF	Comorian Franc
KPW	North Korean Won
KRW	South Korean Won
KWD	Kuwaiti Dinar
KYD	Cayman Islands Dollar
KZT	Kazakhstani Tenge
LAK	Lao Kip
LBP	Lebanese Lira
LKR	Sri Lankan Rupee
LRD	Liberian Dollar
LSL	Lesotho Loti

Abbreviation	Description
LVL	Latvian Lats
LYD	Libyan Dinar
MAD	Moroccan Dirham
MDL	Moldovan Leu
MGA	Malagasy Ariary
MKD	Macedonian Denar
MMK	Myanma Kyat
MNT	Mongolian Tugrik
MOP	Macanese Pataca
MRO	Mauritanian Ouguiya
MUR	Mauritian Rupee
MVR	Maldivian Rufiyaa
MWK	Malawian Kwacha
MXN	Mexican Peso
MYR	Malaysian Ringgit
MZN	Mozambican Metical
NAD	Namibian Dollar
NGN	Nigerian Naira
NIO	Nicaraguan Cordoba
NOK	Norwegian Krone
NPR	Nepalese Rupee
NZD	New Zealand Dollar
OMR	Omani Rial
PAB	Panamanian Balboa
PEN	Peruvian Nuevo Sol
PGK	Papua New Guinean Kina
PHP	Philippine Peso

Abbreviation	Description
PKR	Pakistani Rupee
PLN	Polish Zloty
PYG	Paraguayan Guarani
QAR	Qatari Riyal
RON	Romanian Leu
RSD	Serbian Dinar
RUB	Russian Ruble
RWF	Rwandan Franc
SAR	Saudi Riyal
SBD	Solomon Islands Dollar
SCR	Seychellois Rupee
SDG	Sudanese Pound
SEK	Swedish Krona
SGD	Singapore Dollar
SHP	Saint Helena Pound
SLL	Sierra Leonean Leone
SOS	Somali Shilling
SRD	Surinamese Dollar
STD	Sao Tome And Principe Dobra
SYP	Syrian Pound
SZL	Swazi Lilangeni
THB	Thai Baht
TJS	Tajikistani Somoni
TMT	Turkmenistan Manat
TND	Tunisian Dinar
TOP	Paanga
TRY	Turkish New Lira

Abbreviation	Description
TTD	Trinidad and Tobago Dollar
TWD	New Taiwan Dollar
TZS	Tanzanian Shilling
UAH	Ukrainian Hryvnia
UGX	Ugandan Shilling
USD	United States Dollar
UYU	Uruguayan Peso
UZS	Uzbekistani Som
VEF	Venezuelan Bolivar
VND	Vietnamese Dong
VUV	Vanuatu Vatu
WST	Samoaan Tala
XAF	Central African CFA Franc
XCD	East Caribbean Dollar
XDR	Special Drawing Rights
XOF	West African CFA Franc
XPF	CFP Franc
YER	Yemeni Rial
ZAR	South African Rand
ZMW	Zambian Kwacha