DK Docs

Welcome!

Documentation for www.davidkohen.com and aligned services.

Hi and welcome!

Welcome to the our documentation. On here we will briefly discuss and explain on how to use some of our services.

While we try to keep this page as updated as possible, in cases you might have a custom configuration applied to suit specific needs in which some that is noted here will work differently.

Please note: our full fledged frameworks/platforms have separate docs.

Partners	Description
Nowpayments.io	Guarantees decentral and full control of crypto funds for our framework solution users.
Offertoro.com	A smart and quite profitable service to add to youı casino to gain extra revenue out of your players filling out surveys and installing apps.
HollywoodTV.com	Provides bespoke Live Casino solutions, like you own whitelabel live casino tables & games.

Games API

API Overview

Description of our API service.

Services under 1 API

All the following products work on our unified API service, that means in regards of bet and balance events using same method. We keep it simple, use 1 API for all your iGaming needs.

You only need to integrate our API once and get access to all the following without any API/backend changes:

- All slots
- Live Casino, with exception of custom private Live Casino tables
- All sports & virtual sports services
- Poker

Seamless Balance Integration

Our API works under industry standard *seamless player balance* integration. This means that you host the player balance and tell our API what the player balance is at all times.

In essence our API tells you what is happening on the games, for you then process the results such as adding and subtracting player's balance accordingly on your end.

Extended API

Extended API is same as our regular API, however sending also the exact game data.

Extended API

We can per request enable for you the extended api. The extended api sends all info instead of in REQUEST headers, it sends through JSON Body Format and also includes all the data on API call we get from the individual providers.

This is useful for people that wish tie specific game events to bonus systems, such as challenges systems.

For example, if you want to know the exact cards being played on Blackjack.

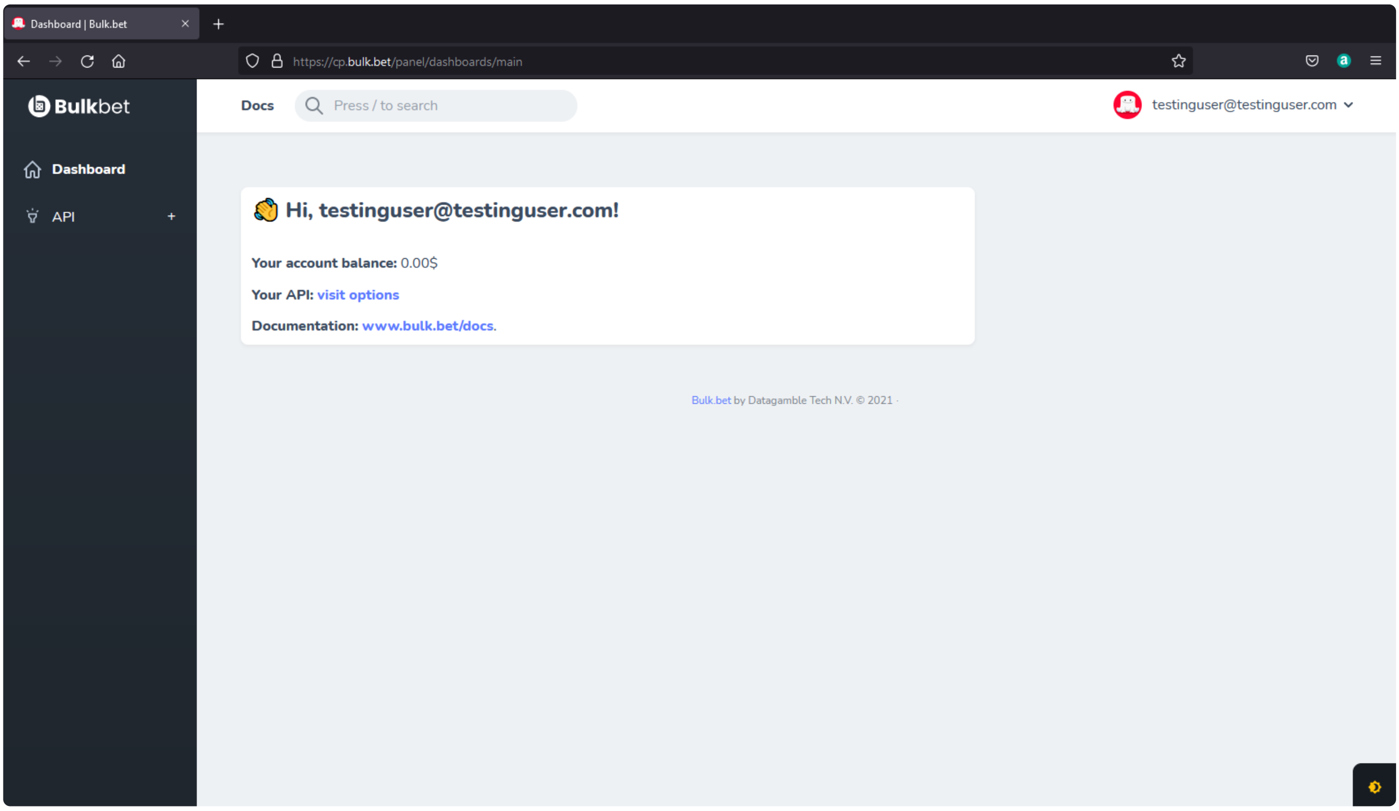
The full data regardless is always available in your "Game History" panel on the back office. We feel like the standard API is more then sufficient for 99% of usecases, but we do want to offer in case is needed a the full data in exact same way as how we receive those.

Please note that the data being sent in extended API is much larger (x20 and in some cases x100 as much weight) and on high volume might impact the game processing on your end.

Setting Callback Endpoint

Here we describe how to set your callback endpoint where we send you all game events and where we expect to retrieve a player's balance.

After getting your credentials, head over to the back-office and login at https://admin.dk.games
You should be presented with the below view, simply expand the menu on the left side:


Dashboard | Bulk.bet
https://cp.bulk.bet/panel/dashboards/main
Bulkbet
Dashboard
API +
Docs
Press / to search
testinguser@testinguser.com

Hi, testinguser@testinguser.com!

Your account balance: 0.00$

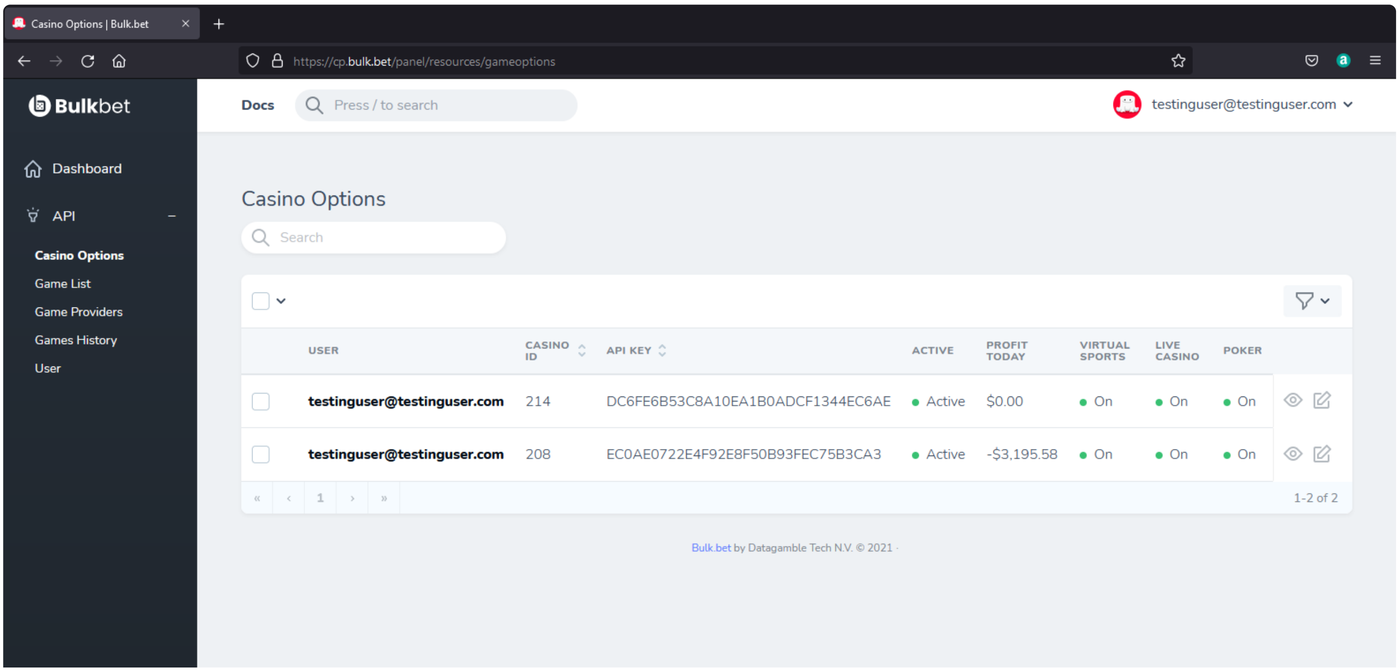
Your API: visit options

Documentation: www.bulk.bet/docs.

Bulk.bet by Datagamble Tech N.V. © 2021 ·

Main Page - Displaying your balance with us (bi-daily charged by GGR)

Now head over to your actual API key & Casino Settings. Feel free to request additional API keys to your account manager before or after your initial setup, under fair-use policy.

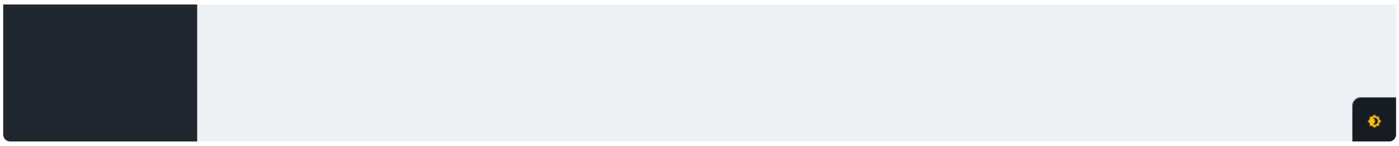


Casino Options

	USER	CASINO ID	API KEY	ACTIVE	PROFIT TODAY	VIRTUAL SPORTS	LIVE CASINO	POKER
□	**testinguser@testinguser.com**	214	DC6FE6B53C8A10EA1B0ADCF1344EC6AE	● Active	$0.00	● On	● On	● On
□	**testinguser@testinguser.com**	208	EC0AE0722E4F92E8F50B93FEC75B3CA3	● Active	-$3,195.58	● On	● On	● On

1-2 of 2

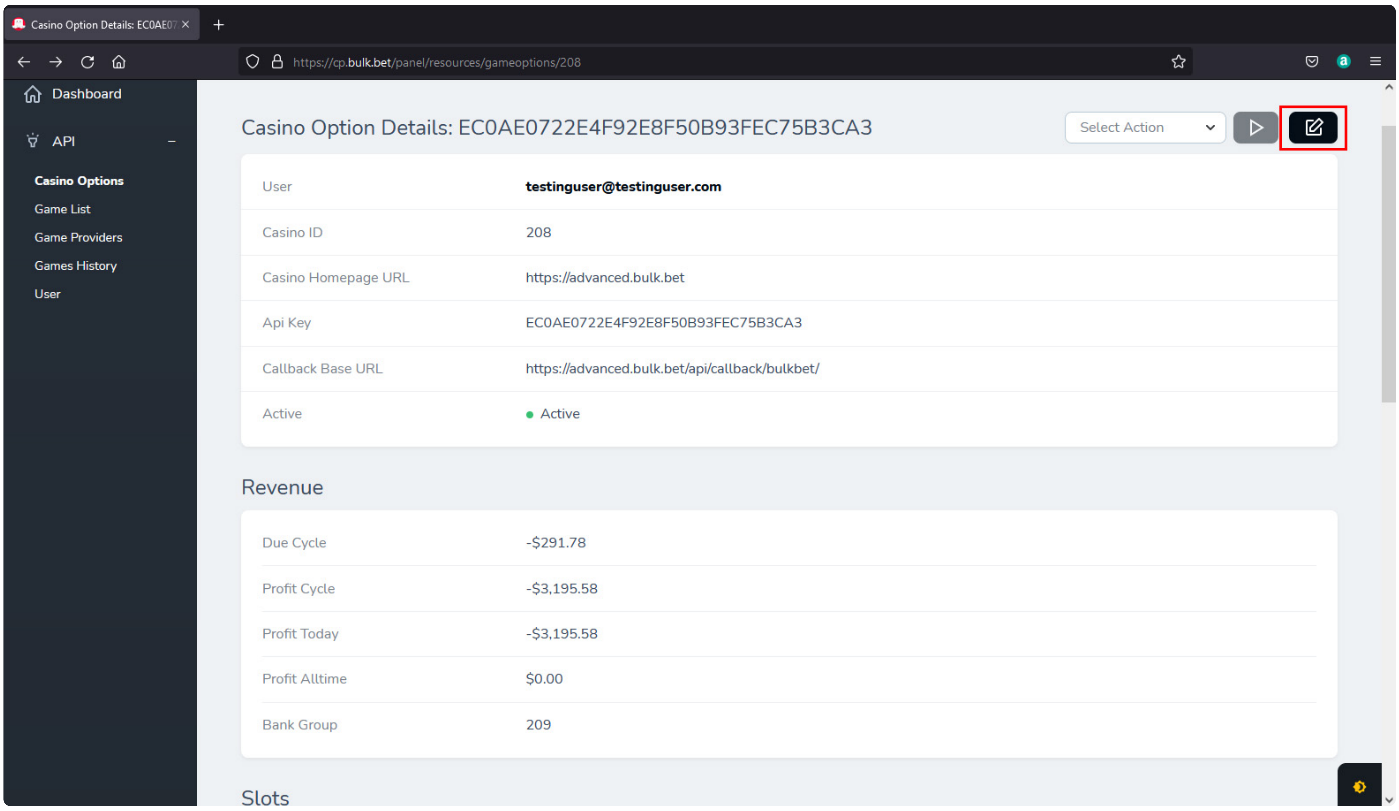
Bulk.bet by Datagamble Tech N.V. © 2021 ·



Configuration per API Key. Check Billing category in docs for info about Profit Cycles.

Now press on the concerning Casino/API Key you would like to make use of. We will be happy to configure and go over the exact configuration for your application.

After selecting your API Key, press the "Edit" button, marked with red rectangle:



Casino Option Details: EC0AE07
https://cp.bulk.bet/panel/resources/gameoptions/208
Dashboard
API
Casino Options
Game List
Game Providers
Games History
User
Select Action

Casino Option Details: EC0AE0722E4F92E8F50B93FEC75B3CA3

User	**testinguser@testinguser.com**
Casino ID	208
Casino Homepage URL	https://advanced.bulk.bet
Api Key	EC0AE0722E4F92E8F50B93FEC75B3CA3
Callback Base URL	https://advanced.bulk.bet/api/callback/bulkbet/
Active	● Active

Revenue

Due Cycle	-$291.78
Profit Cycle	-$3,195.58
Profit Today	-$3,195.58
Profit Alltime	$0.00
Bank Group	209

Slots

Overview showing all your callback URLS on which we will post/request events.

After having edited the callback base url to your webhook URL, you will be able to see the exact URL's we will send service events towards, mainly the "balance request" and "bet post".

Get Games List

API Request to get list of all available games and get the "game_id" that will be used for "Create a Session".

GET https://api.dk.games**/v2/listGames?apikey={APIKEY}**

Request All Games

This endpoint will provide with all games available on your API Key.

Retrieve your API Key from Casino Options section in backoffice.

Parameters

Path

`apikey`	string

Your API Key

Responses

● **200**

Game List successfully retrieved.

›

Example Request:

```
https://api.dk.games/v2/listGames?apikey=DC6FE6B53C8A10EA1B0ADCF1344EC6AE
```

GET https://api.dk.games**/v2/listGames?apikey={APIKEY}&provider={PROVIDER ID}** ˄

Request Games from a Provider

This endpoint will provide all games available for a specific provider

Parameters

Path

`provider`	string

Provider ID - check Provider API request

`apikey`	string

Your API Key

Responses

●

200

Example Request:

```
https://api.dk.games/v2/listGames?apikey=DC6FE6B53C8A10EA1B0ADCF1344EC6AE&provider=netent
```

Returning Balance

Used in Create Session API Methods (to get balance) and on Game Event when a game commences or a player wins.

How we expect you to return Player Balance data

In all cases, you are still in charge of your player's balance/wallets and merely telling us the player balance.

Returning the player balance is the only thing we require you to-do return **on all our API requests we send to you**, without receiving balance in correct format, we will close the player's session or not even be able to start a game session.

The player ID will be in the REQUEST headers when we send you a callback, based on player ID you used on creating session. We can toggle this to BODY headers per request.

This includes returning player's balance on our "bet and result" callback, after you have processed the balance change on your end (you return the balance **after** subtracting/adding to player balance on your end).

We send these to the callback points you have set-up in the back office, please see "Setting Callback Endpoint".

This is how we expect you to return your player's balance in USD value as cents (integer) in JSON format.

```
{
    "status": "ok",
    "result": {
        "balance": 1000,
        "freegames": 0
    }
}
```

In above example a player has 10$ on his balance, you return to us balance "1000" (10$ in cents).

POST https://{CALLBACK_BASEURL}
/{GAME_PREFIX}/balance?playerid={PLAYERID}&currency={SUBCURRENCY}

Balance API Request (we send this to you!)

This is how we request your player's balance on **your endpoint**. This is used mainly for creating the session or when game requires balance to update (without wager being placed).

Please note that the "currency" parameter we send, is only a sub-currency and all games are in USD.

The sub currency is if you have multi-currency balance setup on your casino, for example if you are using crypto, so you can process the balance conversion in the correct player currency.

If you only have 1 currency, you can ignore this parameter.

Parameters

Path

`SUBCURRENCY` string

Optional parameter, if you have set a sub currency to identify on your own end the correct balance type (if you have multiple).

`PLAYERID` string

This is the User ID you used in the "Create Session" API method.

`CALLBACK_BASEURL` string

You can set callback base-url in back office. Please check "Setting Callback Endpoint" in this documentation.

`GAME_PREFIX` string

Each type of game (slots, poker, livecasino, sports) you can configure a prefix. This is in case you would want to receive each type on different endpoints (not required).

Responses

● **200**

This is an example response we expect from you on our API calls.

Example Request:

https://yourcasino.com/api/callback/dk/slots/balance?subcurrency=btc&playerid=6155533344dc226ee94a8579&gameid=starburst_touch

Example Response:
This response we expect you to give, in JSON format. You return the balance in cents and integer (no decimals), in below example you return and tell us the player's balance is 1$ (100 cents).

```
{
    "status": "ok",
    "result": {
        "balance": 100,
        "freegames": 0
    }
}
```

Create a Session

Types of Sessions

Demo Session, Regular Session and Free Spins Session.

Please check the individual pages to get more information.

Create Demo Session

To create a demo session with play money. We do not require you to return balance on this method, as demo/play money is local on provider's side.

API Flow - Demo Game

To start a demo session (with real money balance) flow is as following:

> **1 - Request to us Create Demo Session**
> Once player selects a game on your frontend, you then send a "Create Session" request to our API.

> **2 - Return Player Balance & Take Session URL**

> We return you the demo session URL.

> **3 - Serve Session to your Player**
>
> Serve the Session URL we returned to you in Step 2 within an iframe to your player. The game will now load, displaying standard a 100 DEMO credits as currency.

Create Demo Session API Method

GET https://api.dk.games
/v2/createSession?apikey={APIKEY}&userid=demo&game={GAME_ID}&mode=demo

Create Demo Session

Please note that not all providers support creating demo sessions, mainly live providers do not support demo/free play.

This API method does not require you to return any balance, nor will we send any game events in demo mode to you.

Parameters

Path

`userid` string

Enter any User here, this does not impact but can be handy if you wish to log if a player has tried demo games of a specific games, for you then present a bonus
on that game.

`apikey` string

Your apikey.

`game` string

Use "game_id" to indicate what game to start demo session. You can find the game_id in the games list.

`mode` string

Use "demo" to indicate you wish to start a demo session.

Responses

● **200**

We return Session URL to you

Example Request:

```
https://api.dk.games/v2/createSession?apikey=25DB7D8CC2591A1D7040FABB313BFEF6&userid=demo&
```

Create Regular Session

Describes the Flow and actual API method for creating regular real money sessions.

API Flow - Starting Game

To start a regular session (with real money balance) flow is as following:

> **1 - Request to us Create Session**
> Once player selects a game on your frontend, you then send a "Create Session" request to our API.

> **2 - We request on your callback endpoint the Player's Balance**
> We send a request to your configured callback *balance* URL to retrieve player's USD balance.

> **3 - We generate a session URL**
> After you return the player balance on our request, we then generate and show you *Session URL* to present to player. This session is now linked to by you specified player ID.
>
> ```
>  If we are unable to retrieve player balance, we cannot create a
> session and will return you an error. Errors almost exclusively have
> to do with you returning an incorrect balance or no balance at all,
> in some cases there may be maintenance on provider's side.
> ```

> **4 - Serve Session to your Player**
> Serve the Session URL we returned to you in Step 3 within an iFrame to your player. The game

> will now load, displaying the player balance within the requested game.
>
> This session URL changes on every request. You should repeat this process every time a player starts a game, even if it's the same game. This to preserve session integrity for the player.

GET https://api.dk.games
/v2/createSession?apikey={APIKEY}&userid={PLAYERID}-{SUBCURRENCY}&game={GAME_ID}&mode=r

Create Regular Session API Method

This method creates a real money session.

If you fire this request to us, we will then send you a "balance" API call to your server, requesting balance for the {PLAYERID} you've used in the create session request.

Check the "Returning Balance" page on how to return balance. **Without properly returning balance, session will be unable to create.**

Parameters

Path

`mode` string

Use "real" on this, to flag this session as real money play.

`game_id` string

Use "game_id" as shown on the "Get Games List" method.

`subcurrency` string

If you use multi-wallet balances on your platform, it is for you to identify the correct player wallet/balance. If you only have 1 currency, just use "usd" on this parameter. **This does not affect the actual currency on the game, this is always USD**.

`playerid` string

Your own player identifier, to be used on callbacks so you can identify for who to do any player events.

`apikey` string

Your API Key

Responses

● **200**

Showing you session URL to be used to serve to the player.

● **400**

Example Request:

> https://api.dk.games/v2/createSession?apikey=25DB7D8CC2591A1D7040FABB313BFEF6&userid=613a3f23efeaaf119e012a27-btc&game=100_joker_staxx_ps_html&mode=demo

*Make sure to check **userid** is correctly entered, this consist of: "PLAYERID-SUBCURRENCY". Even if you only have 1 currency on your platform, in that case use "usd", like "player123-usd".*

Create Free Spins Session

Describes how to start up Free Spins sessions.

Free Spins Limitations

Due to abuse in past, by default you are limited to only hand-out free spins on specific providers on which we have special agreements.

You can currently use Free Spins on **Evoplay games**, **without any limitations**. These games are counted as regular spins with regular GGR pricing in your billing.

We can enable per your request free spins on every provider you wish, however, we do limit giving out free spins in that these can only be given to player-ID's that have at least wagered 100$ on any game.

Once you reach 100K$ GGR volume in 30-day period and having paid all your due bills, we can unlock for you to give out free spins on any slotmachine provider without any of above limitations.

API Flow - Launching Bonus Game

To start a free spins session, the flow is as following:

> **1 - Request to us Create Free Spins Session**
> Once player selects a free spins game on your frontend, you then send a "Create Session" request

to our API. (Limited by default to Evoplay Games only)

2 - We request on your callback endpoint the Player's Balance
We send a request to your configured callback *balance* URL to retrieve player's USD balance.

3 - We generate a bonus session URL
After you return the player balance on our request, we then generate and show you *Session URL* to present to player.

This session is now linked to by you specified player ID and free spins will be registered and remain on the specific game, till they have been cleared.

Do not make additional free spins sessions, once player launches into a free spins session, these free spins are registered to the playerID and remain there till cleared.

If a player disconnects or leaves page, these spins will remain and all player needs to do is revisit the game and you can use the "Create Regular Session" API method like if it were a normal game request.

```
 If we are unable to retrieve player balance, we cannot create a
session and will return you an error. Errors almost exclusively have
to do with you returning an incorrect balance or no balance at all,
in some cases there may be maintenance on provider's side.
```

4 - Serve Session to your Player
Serve the Session URL we returned to you in Step 3 within an iFrame to your player. The game will now load and prompt the free spins immediately.

GET https://api.dk.games
/v2/createSession?apikey={APIKEY}&userid={PLAYERID}-{SUBCURRENCY}&game={GAME_ID}&mode=b

Create Free Spins Session API Method

This method creates a free spins session.

If you fire this request to us, we will then send you a "balance" API call to your server, requesting balance for the {PLAYERID} you've used in the create session request.

Check the "Returning Balance" page on how to return balance. **Without properly returning balance, session will be unable to be created.**

Parameters

Path

`freespins_value` string

The bet value per spin, we advise to set this amount to 0.3 ($0.30 per spin).

`superspins` string

Optional parameter, default is "0". Set to "1" to get super free spins.

`freespins` string

Total amount of free spins added

`mode` string

Use "bonus" on this, to flag this session as a free spins session.

`game_id` string

Use "game_id" as shown on the "Get Games List" method - only Evoplay games by default are allowed.

`subcurrency` string

If you use multi-wallet balances on your platform, it is for you to identify the correct player wallet/balance. If you only have 1 currency, just use "usd" on this parameter. **This does not affect the actual currency on the game, this is always USD**.

`playerid` string

Your own player identifier, to be used on callbacks so you can identify for who to do any player events.

`apikey` string

Your API Key

Responses

● **200**

Showing you session URL to be used to serve to the player.

● **400**

Example Request:

> https://api.dk.games/v2/createSession?apikey=250B7D8CC2591A1D7040FABB313BFEF6&userid=6134e75a50a0972bfc0d1c72-usd&game=aeronauts&mode=bonus&freespins=10&freespins_value=1

*Make sure to check **userid** is correctly entered, this consist of: "PLAYERID-SUBCURRENCY". Even if you only have 1 currency on your platform, in that case use "usd", like "player123-usd".*

Bet and Result Event

How we send Bets & Results to your API endpoint

API Flow - Bets & Results

Below is how we generally send the bets & results:

> **1 - Player places a wager**
> Player places a wager on any of our games.

> **2 - We send you the bet request**
> We send the bet request of player with bet and/or win amount and other specifics.

> **3 - You return us the player balance on our bet request**
> After processing the balance modification on your side, you return to use the new player balance, as described in the "Returning Balance" page on this documentation.

> ⚠ Important to note, is that you should always go by the "final" parameter to know when a game is finished.

> Where we are able to, we put the bet & win within a single API request.
>
> However, in some games this is not possible (like live casino, where the outcome result is sent to us much later).
>
> In those cases we will first send a bet callback with parameter "final" set to 0. Then once result is in we will send you a win callback (even if win is 0) with parameter "final" set to 1.
> The round ID is specific and will be the same if we send 2 API requests (bet & win) so you can link these together on your own end, if you wish to do so.
>
> If you do not return player balance for any reason on the callbacks, the game is cancelled and session is closed.
>
> Please make sure to do an additional balance check once bet comes in, in case the bet is bigger then balance you return nothing to us, at which we then close the session.

POST https://{CALLBACK_BASEURL}
/{GAMETYPE_PREFIX}/bet?currency={SUBCURRENCY}&gameid={GAME_ID}&roundid={ROUND_ID}&pl

Bet Callback API Method (we send this to you)

This is how we send bets & results to **your "bet" endpoint**. You then process the bet and return to us the new player balance after deducting and/or adding the win result.

Please note that the "currency" parameter we send, is only a sub-currency and all games are in USD.

The sub currency is if you have multi-currency balance setup on your casino, for example if you are using crypto, so you can process the balance conversion in the correct player currency.

If you only have 1 currency, you can ignore this parameter.

Parameters

Path

`SECURITYHASH` string

This hash can be used to verify callbacks are coming from us, check "Security Hash" page on this docs.

`FINALACTION` string

Indicates if the game is finished. We try to send bet & win within 1 API call where applicable,

if not possible we send 2 API calls (bet and win),
at which this parameter indicates if game is over.

`BONUS_MODE` string

Indicates if being a free spins win/result, after using free spins session API method.

`WIN_AMOUNT` string

Win amount in cents as USD$ value, this amount you add to the player balance on your side.

`BET_AMOUNT` string

Bet amount in cents as USD$ value, this amount you deduct from the player balance on your side.

`PLAYERID` string

Player ID which you entered on creating session

`ROUND_ID` string

Round ID, this is unique per betting round.

`GAME_ID` string

Game that is wager is happening on

`SUBCURRENCY` string

Optional parameter, if you have set a sub currency to identify on your own end the correct balance type (if you have multiple).

`PLAYERID` string

This is the User ID you used in the "Create Session" API method.

`CALLBACK_BASEURL` string

You can set callback base-url in back office. Please check "Setting Callback Endpoint" in this documentation.

`GAMETYPE_PREFIX` string

Each type of game (slots, poker, livecasino, sports) you can configure a prefix. This is in case you would want to receive each type on different endpoints (not required).

Responses

● **200**

This is an example response we expect from you on our API calls.

Example Request:

> ✓ https://yourcasino.com/api/callback/bulkbet/slots/bet?currency=btc&gameid=100_joker_staxx_ps_html&roundid=2:bNTCTJhh6wb4fiJ9&playerid=614b3b90ec9fc17357145d93&bet=40&win=0&bonusmode=0&final=1&sign=db7b7918935aaecf99cf8877eafaaa8f

Example Response:
You would first process and deduct 0.40$ of the player's balance (?bet=40 in above example request), then return new balance.

This response we expect you to give, in JSON format. You return the balance in cents and integer (no decimals), in below example you return and tell us the player's balance is 1$ (100 cents).

```
{
    "status": "ok",
    "result": {
        "balance": 100,
        "freegames": 0
    }
}
```

Security Hash (optional)

Verify the security signature we send on callbacks.

Verify Callbacks are legitimate

To verify all callbacks being legitimate and coming from our server, we have implemented a signature we send along each callback. It is completely up-to you to actually verify the hash on your end, in any case you should implement this only after having completed the rest of integration.

This signature, depending on type of callback is a simple md5 hash, the salt (OPERATOR_SECRET) can be changed to anything within the backoffice, per apikey.

> ⓘ The $OPERATOR_SECRET by default is your Casino ID which you can find in backoffice.

> We advise you strongly to change this to a custom secret, you can do this in the backoffice by editing the OPERATOR_SECRET in your casino configuration.

Bet & Result callbacks security hash

All bet & result security hashes are built and sent under following settings:

```
md5($YOURAPIKEY.'-'.$ROUNDID.'-'.$OPERATOR_SECRET);

// For example:
// md5(DC6FE6B53C8A10EA1B0ADCF1344EC6AE-257E1DNFF3-208);

```

This md5 signature will be present in REQUEST headers but also within the JSON BODY. You take the game `roundid` from the same callback.

You build the md5 hash at your end, then you compare the md5 hash with the signature we sent along and thus verifies if callback is legitimately coming from our API servers.

Balance callbacks security hash

If you wish to secure the balance callbacks (however posing no threat other then showing a player's balance):

```
md5($APIKEY.'-'.$PLAYERID.'-'.$OPERATOR_SECRET)

// For example:
// md5(DC6FE6B53C8A10EA1B0ADCF1344EC6AE-Player1491-208);
```

You build the md5 hash at your end, then you compare the md5 hash with the signature we sent to verify the origin.

> You can also implement IP filter to only allow access from our API servers on your endpoints.
>
> However our API IP's do change on occasion and this can happen at any point without notice, so we advise to build in a security check using the security hash.

> To get an updated IP list of all our API servers, ask your account manager.

Rollback Method

Rollback Method

We will rarely - in cases where transaction can not properly be processed on our side, send you a rollback callback.

We send this as a regular bet/win but with game_id "rollback" and credit either player's bet or subtract win, depending on type of error.

You should log this transaction locally so you can review the rollback by looking in JSON body where we will place reason of rollback (not always).

Moreover, you can find this in your game history in backoffice by searching "rollback".

All you need to do on your end is to treat and process this as a regular game transaction and debit/credit like if it would be a normal game.

Example Snippets

PHP Example Application

This is a PHP-only application to get a better understanding of our API. All games are functional on this application, writing "callbackLogs" and retrieving and changing balancefrom balance.txt.

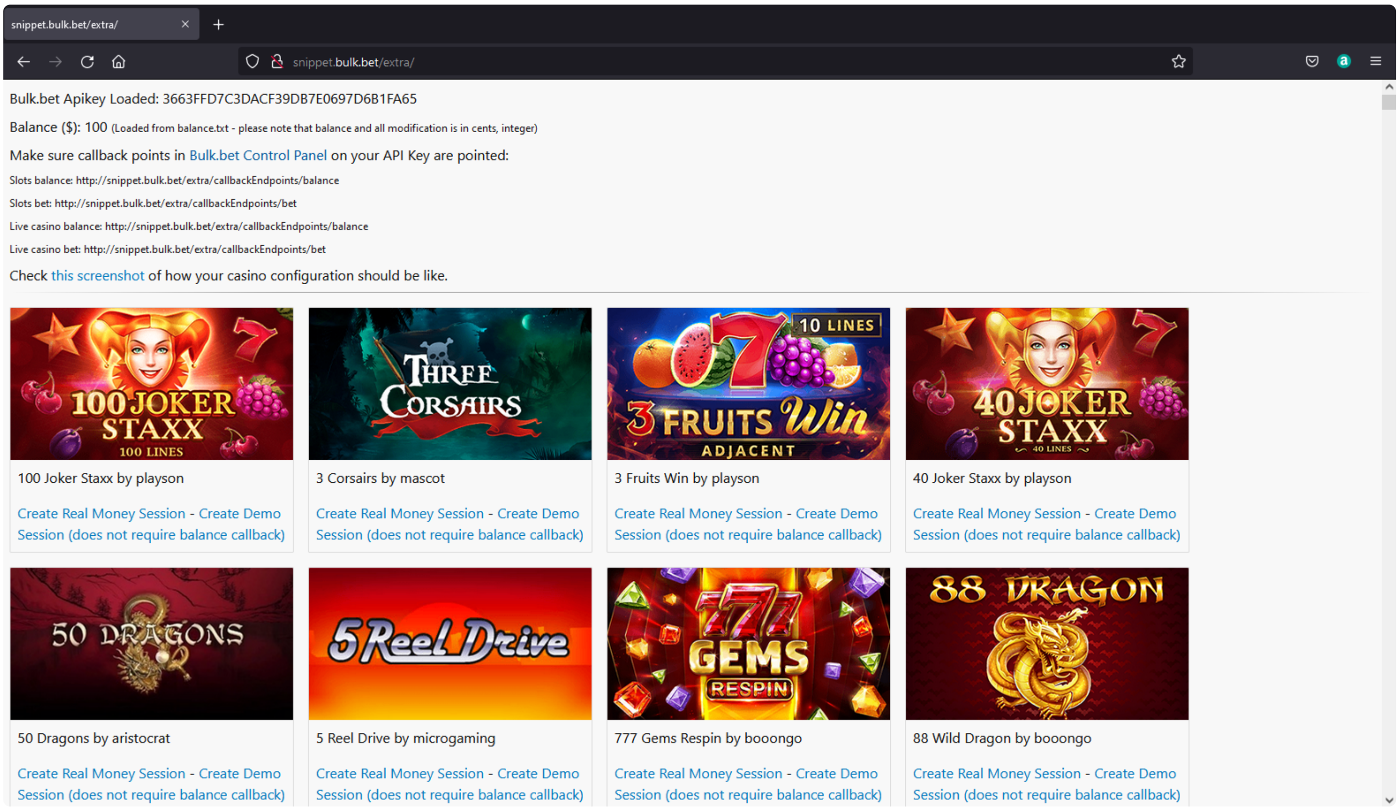
Getting Started with Example Application

You can download the PHP Example application straight from Github:
https://github.com/isaackohen/davidkohen-example-app

Simply CHMOD 755 (preferably 777) the application and enter your API Key within index.php.
The homepage is rendered using the `ui/index.htm` template and showcases how to make use of our image CDN.

This application is only for testing purposes and is not safe to implement in-to production, but does give a deeper understanding in our API quite easily, as all that is required is PHP and a webserver.

It retrieves the gameslist live from our API using the "Get Games List" API method.



Bulk.bet Apikey Loaded: 3663FFD7C3DACF39DB7E0697D6B1FA65

Balance ($): 100 (Loaded from balance.txt - please note that balance and all modification is in cents, integer)

Make sure callback points in Bulk.bet Control Panel on your API Key are pointed:

Slots balance: http://snippet.bulk.bet/extra/callbackEndpoints/balance

Slots bet: http://snippet.bulk.bet/extra/callbackEndpoints/bet

Live casino balance: http://snippet.bulk.bet/extra/callbackEndpoints/balance

Live casino bet: http://snippet.bulk.bet/extra/callbackEndpoints/bet

Check this screenshot of how your casino configuration should be like.

100 Joker Staxx by playson

Create Real Money Session - Create Demo Session (does not require balance callback)

3 Corsairs by mascot

Create Real Money Session - Create Demo Session (does not require balance callback)

3 Fruits Win by playson

Create Real Money Session - Create Demo Session (does not require balance callback)

40 Joker Staxx by playson

Create Real Money Session - Create Demo Session (does not require balance callback)

50 Dragons by aristocrat

Create Real Money Session - Create Demo Session (does not require balance callback)

5 Reel Drive by microgaming

Create Real Money Session - Create Demo Session (does not require balance callback)

777 Gems Respin by booongo

Create Real Money Session - Create Demo Session (does not require balance callback)

88 Wild Dragon by booongo

Create Real Money Session - Create Demo Session (does not require balance callback)

Instantly run a working test application with support for all games.

Billing

Billing Cycles

Quick and clear insight in how we calculate the charges and GGR (gross game revenue) on gains through each individual provider

After you have logged in, expand the side menu and head over to the Casino Options section.

This section contains your callback endpoints and shows your profit amount today, profit cycle amount, profit total and *estimated due amount.*

We bi-daily charge the GGR (*gross gaming revenue*) from your account balance, if you have made a loss we will not charge you anything.

> GGR is a term used in iGaming and represents the profit you have accumulated through games by specific provider.
>
> If you have made 100$ profit on a specific gaming provider, with a 10% GGR rate means you will have to pay 10$ on billing cycle. Each provider has their own individual pricing, which you can find on the game providers section.
>
> If you have made a loss using a specific game provider, we will charge you nothing.

Please note that as we do offer you direct to source access towards each provider, while our prices are by far most competitive, pricing is calculated on a *per provider basis*.

The *estimated due amount* on current billing cycle is merely an indication of upcoming costs and can differ in some cases to the actual bruto GGR amount.

You can always see past charges and of course review and download each and every game transaction that has gone through our systems to verify.

Let's take this example:

- You have made a net loss of games played through Evolution Gaming of **-100,00$**.
- You have made a net profit of games played through NetEnt of **+50,00$**.

In above example we would charge you the on-going %GGR rate individually and charge you the GGR pricing of NetEnt (*50$ * ggr%*) in above example, while charge nothing for Evolution Gaming.

> Please note that some providers require a minimum GGR revenue per month, at which we do not charge you anything bi-daily on the specific provider till you have reached the minimum GGR monthly amount.
>
> We bill the first month's minimum from and to the 1st of every month from your account balance. The minimum has to be paid in advance to make use of such provider.

We do not invoke any exclusivity requirements, this means you can use our API in conjunction with any other iGaming party.

If you have sufficient volume and you or a provider request us to mediate in individual price agreements directly with the provider or specific promotions, we will always assist in such.

We usually do pro-actively contact you, as our partnered providers generally are very interested in our customers to grow their iGaming activities and are open for custom promotions.

Account Balance

Upload your account balance

Uploading Account Balance

In order for API to be active, we require you to have more then 0$ account balance. If minus, system automatically gives you 24 hours to make sure to become positive again, you will receive an automatic e-mail.

Pay links are setup individually per operator upon activation, depending on choice of payment and are all instant (with exception of Cash Deposit).

We allow **Bank Transfer** (from 25K$ and above), **Creditcard Payment** (from 1K$ up to 10K$), **Cryptocurrency** (from 1K$ and above), **Cash Deposit** (by DHL with insurance from 5K$ up to 10K$).

Bank Transfer is to our CHASE Bank account in USA (worldwide) or Instant Bank Transfer to our account in U.K. (only applicable if you send from U.K. bank account).

Extension

In the case of payroll difficulties please immediately contact your account manager, based on your previous record, we can give payment extensions and are happy to work with you to overcome any difficulties.