

Dragon Gaming TM

Integration Guide

Contents

▶ Introduction	3
▶ Glossary	3
▶ Integration Process	4
▶ Integrate DragonGaming™ Backoffice	5
▶ Customer Environment URLs	5
▶ Customer Test Environment URLs and Parameters	5
▶ Production Environment URLs and Parameters	5
▶ Casino Lobby Design and other functionality	6
▶ Login and Register Users	6
▶ Currency and Coin Value for Games	6
▶ Jackpots	6
▶ Display Game History	6
▶ Unfinished Game Rounds	8
▶ Publish Unfinished Game Rounds and Session Timeouts Terms and Conditions	8
▶ White Label Programs	8
▶ Register White Label	8
▶ Player Username Prefix	9
▶ Bonus Programs	9
▶ Player Currency Conversion	10
▶ Integration Reference	11
▶ REST API	11
▶ Operator and Merchant Identification	11
▶ Response Caching	11
▶ Response Formats	11
▶ Errors / Handle Exceptions	11
▶ Supported Countries, Currencies and Languages	12

Introduction

As a game provider, our objective is to deliver exceptional games and experiences to our customers and their players. We would like to make the entire journey for our customers as smooth and easy as possible. Platform integration is a significant part of that journey and we hope that this document will help. This integration guide will be reviewed and revised on a regular basis to ensure we provide you with the best and fully optimised method of integration to you. Please feel free to share any feedback that you might have on this document so that we can review and update the document on the basis of your feedback.

This guide provides information on how to integrate our games with an Operator's systems. It should be followed in conjunction with our API document, Chronos API Document.

This guide consists of the following parts:

- ▶ Overview of the integration process and detailed information about the main activities in the integration process.
- ▶ Detailed description of activities to be performed during the integration process.

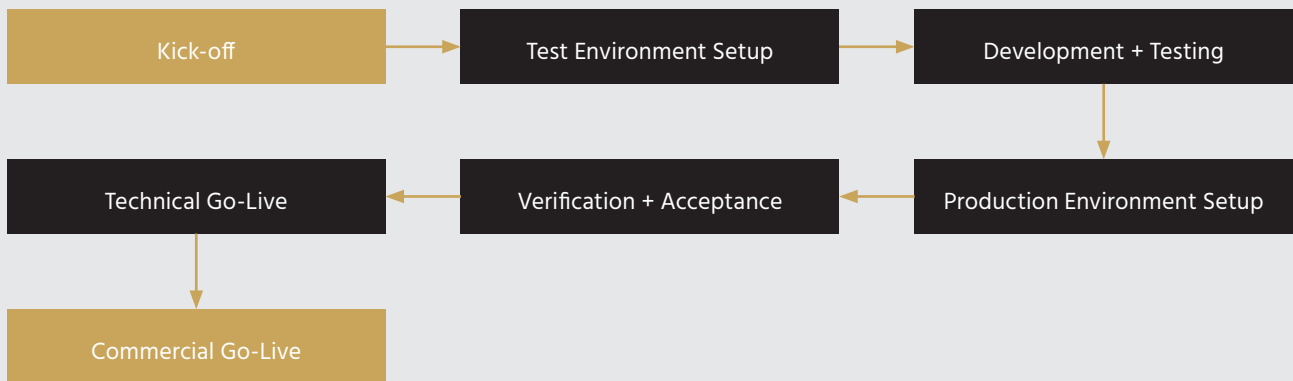
Glossary

Terms	Description
Operator	Anyone that owns / operates a website / front-end
RGS	Remote Game Server (by provider)
Provider	DragonGaming™ / Gaming software provider
Session ID / Token	Session ID of the player
API Key	Unique key provided by provider to each Operator
Front-end / Site / White Label / Platform	Operator's website on which games are displayed
Game Lobby	Area where games are shown on Operator's website
Operator ID	Unique Casino ID

Integration Process

The following steps will aid in the integration between DragonGaming™ and new Operators.

The integration of a new Operator is a process triggered by signing an agreement. The process ends when the new integration is live in production and DragonGaming™ games are available on the Operator platform. The main activities in the integration process are shown below.



Workflow

	Actions in the process between DragonGaming™ and the Operator	Who's Responsible
1	<ul style="list-style-type: none"> ▶ DragonGaming™ assigns Integration Manager. ▶ DragonGaming™ schedules a technical start-up/process overview meeting with the Operator. 	integration@dragongaming.com
2	<ul style="list-style-type: none"> ▶ DragonGaming™ setup a test environment and provides access to test environment for Operator. 	DragonGaming™ Integration Manager
3	<p>System Integration Process: Operator starts the system integration by implementing the API methods and steps mentioned in this guide.</p>	Operator
4	<ul style="list-style-type: none"> ▶ Operator tests and verifies integration with the support from DragonGaming™ Integration Manager. ▶ DragonGaming™ provides access to production environment for technical go-live (for regulated markets, this requires a licence first). 	Operator + DragonGaming™ Integration Manager
5	Operator opens to real money players. System is commercially live!	Operator + DragonGaming™ Integration Manager

Integrate DragonGaming™ Backoffice

This section provides information on how to set up the various casino functions required when integrating DragonGaming™ Backoffice with your system.

Customer Environment URLs

The following tables provide the URLs and parameters for test environment and production environment.



The Operator ID is unique per Operator and is used in test and production environments. It is an integer assigned by the provider's platform.

Customer Test Environment URLs and Parameters

Test Backoffice	Description
Backoffice https://test-cms.dragongaming.com	The end point for Backoffice of DragonGaming™
Test API https://test-api.dragongaming.com/v1	Base URL for Test API Environment
Game Load URL https://test-games.dragongaming.com/game_launcher.php	Game launcher URL from which game UI, graphics, sounds, animations are loaded.

Production Environment URLs and Parameters

Backoffice	Description
Backoffice https://cms.dragongaming.com	The end point for Backoffice of DragonGaming™
Test API https://test-api.dragongaming.com/v1	Base URL for Test API Environment
Game Load URL https://games.dragongaming.com/game_launcher.php	Game launcher URL from which game UI, graphics, sounds, animations are loaded.

Casino Lobby Design and other functionality

The Operator is responsible for their Casino lobby design. Typically, this provides following functions:

- ▶ Cashier function through which players manage the funds in their accounts
- ▶ Links that launch casino games
- ▶ Information about bonuses, promotions, chips and tournaments
- ▶ Wallet functionality, deposits module and withdrawals
- ▶ Login and registration of users

Login and Register Users

All new players registered on an Operator's website will be automatically registered on the RGS when the player launches any DragonGaming™ games. The Operator is required to make an API call to our RGS API, just before the player is presented with game UI. The RGS then registers player details if not registered and ignores if already registered.

Further details can be found in The DragonGaming™ Chronos API Document.

Currency and Coin Value for Games

We support 41 currencies. A complete list of currencies is available in Appendix A of this guide. Our Backoffice is available in 3 different currencies – EUR, GBP and USD.

For each supported currency, coin values can be chosen for each game that do not normally need to be changed. For information about the currency values (for example, chip values or coin values) per game, contact your Technical Integration Manager.

Contact our Customer Support Team who can advise you on the values that can be configured and who can change the values for you **support@dragongaming.com**

Jackpots

As of now Jackpot games are not available on DragonGaming™ platform. We will update this section after we add jackpot games to our games portfolio.

Display Game History

A common service our Operators offer their players is the ability to list historical data about their latest game rounds. This helps build trust towards the Operator and can reduce Customer Support enquiries. By default, the game history feature is not enabled. You may discuss this further with your Technical Integration Manager during the integration phase. If you wish to enable the game history feature at a later date, please contact **support@dragongaming.com**

Game History

Players can view their game history using the 'Game History' option located in game UI. Upon clicking on the button, the player will be redirected to a URL with the parameters outlined below. Only the most recent 100 game rounds will be shown to the player. The player can access game rounds beyond the most recent 100 from their account which would be made available by the Operator to the player. The outcome of each base game round can be viewed in a graphical representation. Bonus games details can be viewed in tabular text format.

```
https://games.dragongaming.com/games/player_game_history.php?sessionId=<sessid>&lang=en&playerId=<player_id>&gameType=<slots>
```

Parameter	Description
sessionId	Session ID of the player
gameType	Type of the game. Ex: slots
lang	Locale of the player
playerId	Unique player ID assigned to the player

We will provide the API for game history if this is required for displaying information in your front-end. The In-game history page and main game results will be displayed in text and graphics format. Feature round results will be displayed in text format.

Parameter	Description
lang	A two-character ISO 639 language code in lower-case For Example: en
sessionId	The player's current session ID that was previously returned by loginUser-Detailed
gameRoundId	The game round ID
gameType	Game Type



The game history can be referenced from the games or directly from the casino lobby by entering any of the history URLs.



```
https://games.dragongaming.com/games/game_round_history.php?gameRoundId=<DragonGaming gameRoundId>&sessionId=<sessid>&lang=en&gameType=<GameType>
```

Unfinished Game Rounds

There are ways of displaying and directing the players to game rounds that they never completed, to allow the players to finish the game rounds.

An API will be made available to the Operator if they require access to unfinished game rounds. This API will fetch all unfinished rounds for the last 30 days.



There is also an automated job available in the DraginGaming™ Backoffice to finish unfinished game rounds. This job is disabled by default but may be enabled by DragonGaming's™ Customer Support support@dragongaming.com. If the job is enabled, the default setting is to finish game rounds that have been open for 6 months.

Publish Unfinished Game Rounds and Session Timeouts Terms and Conditions

Players should refer to the Operator site for terms and conditions for the handling of the unfinished game rounds and session timeouts.

White Label Programs

Operators who provide White Label platforms can use Backoffice's admin tool to generate statistics and reports on their White Labels.

In RGS, each front-end / white label is identified with a unique front-end ID or site ID. Every front-end is associated with an Operator. Each Operator is assigned a unique Operator ID. Any Operator can have any number of front-ends. Reports of each front-end are tracked separately.

In summary, every player belongs to a specific front-end and every front-end belongs to a specific Operator.

In order to make API calls to RGS API, each front-end is assigned a unique API_KEY that should be sent in the request parameters. Upon API_KEY validation, list of games enabled for that specific front-end will be sent back to the API caller.

Register White Label

Each White Label **MUST** have a unique identification code.



- ▶ A unique API key is assigned to every front-end ID (used to access the REST API).
- ▶ Every player is tagged to a front-end ID and every front-end is tagged to an Operator ID.

To register a new front-end with RGS, for the first time, the Operator is required to provide the basic details of the front-end. Below are the non-exhaustive parameters.

Parameter	Description
sessionId	Session ID of the player
gameType	Type of the game. Ex: slots
lang	Locale of the player
playerId	Unique player ID assigned to the player

Player User Name Prefix

Some Operators have several casino websites with separate player databases, but all these interact with the same Backoffice. To be able to see from which casino site a player comes from, we recommend that you have several merchant accounts that include the different casino specific White Label codes in the account name, and then add a merchant account prefix to the Usernames.

For Example:

Two players with the same **userName** xyzPlayer (one from the casino CasinoSite1 and the other from the casino CasinoSite2) get the following user names in Backoffice, CasinoSite1xyzPlayer and CasinoSite2xyzPlayer.

Bonus Programs

As a game provider, along with real cash mode, we support the games with various bonus programs such as bonus chips, free rounds and other promotional bonus modes as long as there is a clear indication from Operator to provider about the mode which player is playing with. Demo mode (fun mode) will also be available.

In our RGS, we represent each mode with a unique amount type which is an integer. For example, when the player is playing with real cash mode, we represent cash mode with amount_type 1, bonus mode with 2, free rounds with 3 and fun mode with 4 and so on. This integer value of mode will be part of the URL parameters of the game launch URL. More details on each mode and their representation is available in DragonGaming™ Chronos API Document.



https://games.dragongaming.com/game_launcher.php?amountType= <amount_type> &sessId=<sess_id>...

Tournaments

RGS supports tournaments as well. RGS differentiates the tournaments with separate amount_type as explained above.

Free Rounds Bonus Programs

Free rounds bonus programs enable players to try out specific games without having to pay, and free rounds are used to reward the players and motivate them to play. This is not the same as playing a game in 'Play for fun' mode as there may still be a cost involved for free rounds in terms of whether the player receives any winnings as a bonus or real money at the end of the free rounds.

Awarding players and managing free rounds is done at Operator end.

Currency Exchange Rates

RGS does not consider currency exchange rates of any gaming transactions (debits and credits) as all transactions are carried out in player's currency. For RGS, transaction values are just numbers. Game client is capable of handling multiple currencies when it comes to display of various currencies. All game logs are maintained in player's currency. Player's balance, bet amount and win amounts are shown in player currency.

Currency exchange rates are applied in Backoffice. Backoffice is available in one of these three currencies - EUR, GBP, USD. The Operator must choose a currency during the integration phase and the Backoffice currency will be assigned based on that decision. A third party service, <https://fixer.io> is used for BO currency exchange rates. Exchange rates are applied twice a day. Every day at 10 AM GMT and 4 PM GMT.

Player Currency Conversion

After migrating from a single currency casino to a multi-currency casino you can convert a player to a different currency, if required. This can be discussed during the technical integration phase. The following would be the typical way of handling Player Currency Conversion.

When you convert a player to a new currency the following occurs:

- ▶ Operators must deactivate the player's single currency account and the player must be provided with a new Username. The player's Username must be unique. Otherwise, if the old Username is passed on to gameLaunch API, the RGS will not register the player details as the name is already present. In case of a new Username, the RGS registers the player details.
- ▶ Operators must convert current player limits to the new currency using the current exchange rate. Expired player limits (valid until this point), are not converted.
- ▶ Need to ensure unfinished game rounds are completed, residual bets are paid out to the old account, and any game reservations are resolved.
- ▶ The player's bonuses are forfeited if available.

Integration Reference

APIs provide the main integration points for your casinos. Methods in these APIs are used to manage a wide range of casino functions such as `getGames()`, `registerSessionData()`, `getRoundData()` etc.

RGS may make API request such as `getPlayerData()`, `getPlayerBalance()`, `debitAmount()`, `crediteAmount()` etc.

This section provides key information about the APIs used for integration and describes the documents and resources available to assist in integration with Backoffice.

REST API

DragonGaming™ provides a public REST API and Backoffice that is a standard web service interface.

Operator and Merchant Identification

All sensitive calls to Backoffice require the operator's system to provide a Username and Password .

All calls to RGS / Chronos API requires the operator to send valid API_KEY which will be provided to the operator before integration.

Response Caching

For performance reasons, we cache response data for as long as possible to reduce the number of calls you need to make to this service.

Response Formats

JSON would be the response format. The ordering of elements is not fixed and can vary from call to call. You must always check that you are reading the correct element (that is, you must not rely on the order of elements). For instance, the order of the key-value pairs from Array methods is not fixed.

Errors / Handle Exceptions

Exception handling and error codes are described in the API documentation.

The operator is responsible for making sure that all exceptions thrown by the RGS API are handled in an appropriate manner. More details on Errors and exceptions can be found in Chronos API document.

Supported Currencies

Listed below are all currencies currently supported for display by DragonGaming™.

Country Name	Currency
Argentina	ARS
Australia	AUD
Brazil	BRL
Bulgaria	BGN
Canada	CAD
China	CNY
Croatia	HRK
Czech Republic	CZK
Denmark	DKK
European Union	EUR
Georgia	GEL
Ghana	GHS
Hong Kong	HKD
Hungary	HUF
Iceland	ISK
Indonesia	IDR
Japan	JPY
Lithuania	LTL
Malaysia	MYR
Mexico	MXN
New Zealand	NZD
Norway	NOK

Peru	PEN
Philippines	PHP
Poland	PLN
Romania	RON
Russia	RUB
Singapore	SGD
South Africa	ZAR
South Korea	KRW
Sweden	SEK
Switzerland	CHF
Taiwan	TWD
Thailand	THB
Turkey	TRY
Ukraine	UAH
United Kingdom	GBP
🔗 Bitcoin	BTC
🔗 Bitcoin Cash	BCH

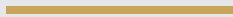
Please Note:

Operators are responsible for conversions, and must then provide equivalent values in USD, EUR or GBP to the Game Engine.

Supported Languages

Below are a list of all languages currently supported by DragonGaming™:

Languages
Bulgarian
Danish
English



Finnish
French
German
Indonesian
Italian
Japanese
Korean
Latvian
Lithuanian
Malaysian
Norwegian
Polish
Portuguese
Romanian
Russian
Serbian
Simplified Chinese
Spanish
Swedish
Thai
Traditional Chinese
Turkish
Vietnamese