**RiverPlay Platform API**

Platform Integration document

Module : Responsible Gaming

**Abstract**

The document is aimed at exposing the Responsible Gaming APIs for platform integration.

# RiverPlay platform APIs

## Introduction

The RiverPlay RG APIs are aimed at exposing the functionalities and services that are needed ensure that players have a safe and fair gaming experience and are protected from the adversity of excessive gambling. All the APIs are exposed through RESTful API interface for integration by the partners.

## Security and access control for the APIs

Oauth 2.0 client\_credentials layer has been implemented on top of the APIs to authenticate and authorise the access to APIs.For more details on that see River fund-b2b-api.docx.

## APIs

Below are the list of APIs exposed for Responsible Gaming:

1. setDepositLimits
2. getDepositLimits
3. setRGCooloffPeriod
4. getRGCooloffPeriod
5. setRGClosed
6. getRGClosedDetails

In all the APIs :

a .Request Method is ‘POST’

b. Header is

Content-Type : application/json,

Accept **:** application/json

### **1. setDepositLimits**

**INTRODUCTION***:*

This API is used to set the deposit limits for the user. Four types of deposit Limits can be set namely daily, weekly, monthly and lifetime.

**URL :** http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/setRGDepositLimit

**METHOD:** POST

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920",

"currency":"EUR",

"dailyLimit":5,

"weeklyLimit":15,

"monthlyLimit":50,

"lifetimeLimit":100,

"agentName":"John",

"comments":"COMMENTS"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

"errorDescription":null,

"ecrExternalId":"3821403877040920"

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Field Name | Field Type | Mandatory | Value/Comments |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| 6 | currency | String | Yes | ISO 4217 Currency Code |
| 7 | dailyLimit | Integer | No |  |
| 8 | weeklyLimit | Integer | No |  |
| 9 | monthlyLimit | Integer | No |  |
| 10 | lifetimeLimit | Integer | No |  |
| 11 | agentName | String | No | agent who is sending this request |
| 12 | comments | String | No |  |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE/PENDING” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |

### **2. getDepositLimits**

**INTRODUCTION***:*

This API is used to get the deposit limits of the user. Four types of deposit Limits are daily, weekly, monthly and lifetime.

**URL :** http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/getRGDepositLimits

**METHOD:** POST

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

"errorDescription":null,

"ecrExternalId":"3821403877040920",

"dailyLimit": 500,

"weeklyLimit":1000,

"monthlyLimit":1500,

"lifetimeLimit":5000

|  | Field Name | Field Type | Mandatory | Value/Comments |
| --- | --- | --- | --- | --- |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |
| 5 | dailyLimit | Integer | Yes | 0 if not set. in minimum denomination |
| 6 | weeklyLimit | Integer | Yes | 0 if not set. in minimum denomination |
| 7 | monthlyLimit | Integer | Yes | 0 if not set. in minimum denomination |
| 8 | lifetimeLimit | Integer | Yes | 0 if not set. in minimum denomination |

### **3. setRGCooloffPeriod**

**INTRODUCTION***:*

This API is used to set Cool Off Period for a user.

**URL :** http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/setRGCooloffPeriod

**METHOD:** POST

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920",

"timeInSeconds":432000,

"agentName":"system",

"comments":"comments"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

"errorDescription":null,

"ecrExternalId":"3821403877040920"

"endData":null

"cooloffStatus":null

|  | Field Name | Field Type | Mandatory | Value/Comments |
| --- | --- | --- | --- | --- |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| 6 | timeInSeconds | Integer | Yes |  |
| 7 | agentName | String | No |  |
| 8 | comments | String | No |  |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |
| 5 | endDate | Integer | Yes | null |
| 6 | cooloffStatus | String | Yes | null |

### **4. getRGCooloffPeriod**

**INTRODUCTION***:*

This API is used to check if the user is in Cool Off Period.

**URL :** http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/getRGCooloffPeriod

**METHOD:** POST

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

"errorDescription":nul,

"ecrExternalId":"3821403877040920"

"endDate":null,

"cooloffStatus": "inactive"

|  | Field Name | Field Type | Mandatory | Value/Comments |
| --- | --- | --- | --- | --- |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |
| 5 | endDate | Integer | Yes | Unix Time if the user is in cool off period, null otherwise |
| 6 | cooloffStatus | String | Yes | “inactive”/“active |

### **5. setRGClosed**

**INTRODUCTION***:*

This API is used to set the RG Closed flag of the user i.e. the closure option for the players when they want permanent exclusion or long term break.

**URL :** http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/setRGClosed

**METHOD:** POST

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920",

"rgClosed":true,

"agentName":"John",

"comments":"COMMENTS"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

"errorDescription":null,

"ecrExternalId":"3821403877040920"

"rgClosed":true

|  | Field Name | Field Type | Mandatory | Value/Comments |
| --- | --- | --- | --- | --- |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| 7 | rgClosed | Boolean | Yes | Set true if the player is to be permanently excluded. |
| 8 | agentName | String | No | agentName |
| 9 | comments | String | No |  |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |
| 5 | rgClosed | Boolean | Yes | Updated flag if it SUCCESS otherwise NULL |

### **6. getRGClosedDetails**

**INTRODUCTION***:*

This API is used to check if the player has been on permanent exclusion or long term break.

**URL :** <http://stress-vendor1.cloudapp.net/gateway-inbound-1.0/v1/ecr/rg/getRGClosedDetails>

**METHOD: POST**

**HEADER :**

**Content-Type :** application/json,

**Accept :** application/json

**REQUEST OBJECT**:

"accessToken":"5c684e6af56d1624dcf5dfd457a696be",

"partnerID":"river",

"labelID":"river",

"productID":"POKER",

"ecrExternalId":"3821403877040920"

**RESPONSE OBJECT** :

"status": "SUCCESS",

"errorCode" : null,

“errorDescription”:null,

“ecrExternalId”:”3821403877040920”,

“rgClosed”:true

|  | Field Name | Field Type | Mandatory | Value/Comments |
| --- | --- | --- | --- | --- |
| **Request** | | | | |
| 1 | accessToken | String | Yes | CLIENT\_CRETENTIALS |
| 2 | partnerID | String | Yes | PartnerID for the Client/partner |
| 3 | labelID | String | Yes | labelID of the product |
| 4 | productID | String | Yes | productID of the product e.g. “POKER” |
| 5 | ecrExternalId | String | Yes | unique ID for the user |
| **Response** | | | | |
| 1 | status | String | Yes | “SUCCESS”/”FAILURE” |
| 2 | errorCode | String | Yes – if status : SUCCESS | Error code providing info on why the request failed.  eg. technical\_error,database\_error,...  Refer Annexure list of ErrorCodes |
| 3 | errorDescription | String | Yes – if status : FAILURE | English Description for the error code |
| 4 | ecrExternalId | String | Yes – if status : FAILURE | External ID for the user |
| 5 | rgClosed | Boolean | Yes | Flag is true when player is permanently excluded |

### **List of ErrorCodes :**

INVALID\_REQUEST : The request is not a valid one. To check the API list accessible to partner again.

TECHNICAL\_ERROR : Non specific technical error.

DATABASE\_ERROR : Technical error, to do with database

ECR\_DOESNOT\_EXIST : The customer account requested is not found.

INVALID\_TOKEN : Auth token has expired or is not valid.