

# Wallet Integration

## Introduction

This document describes integration between *Adapter* and an *Wallet* via *seamless API*.

The key words `MUST`, `MUST NOT`, `REQUIRED`, `SHALL`, `SHALL NOT`, `SHOULD`, `SHOULD NOT`, `RECOMMENDED`, `MAY`, and `OPTIONAL` in this document are to be interpreted as described in [RFC 2119](#).

## Changelog

Version	Date	Changes
1.0.10	31.11.2020	<ul style="list-style-type: none"><li>Added <i>Adapter's</i> playerId</li></ul>
1.0.9	06.09.2020	<ul style="list-style-type: none"><li>Moved <code>currency</code> to <code>/launch</code></li><li>Added explanation of transactional flow</li></ul>
1.0.8	26.08.2020	<ul style="list-style-type: none"><li>Added <code>nativeId</code></li><li>Added <code>provider</code> and <code>roundFinished</code> to <code>/transaction</code> request</li><li>Added <code>/replay</code> launcher</li><li>Reverted <code>wallet</code> param</li></ul>
1.0.7	03.08.2020	<ul style="list-style-type: none"><li>Added launch mode</li><li>Changed <code>user</code> to <code>player</code></li></ul>
1.0.6	20.07.2020	<ul style="list-style-type: none"><li>Changed paths to transaction requests</li></ul>
1.0.5	05.07.2020	<ul style="list-style-type: none"><li>Removed <code>wallet</code> param</li></ul>
1.0.4	23.06.2020	<ul style="list-style-type: none"><li>Separated <i>Wallet</i> and <i>Operator</i></li><li>Added <code>provider</code> param to the launcher</li></ul>
1.0.3	19.06.2020	<ul style="list-style-type: none"><li>Added <code>jackpotAmount</code>, <code>promotionType</code> and <code>promotionId</code> to <code>/transaction</code> request</li><li>Added Launcher section</li></ul>
1.0.2	18.06.2020	<ul style="list-style-type: none"><li>Authorization header is now <code>Bearer &lt;token&gt;</code> instead of <code>&lt;token&gt;</code></li><li>Removed <code>USER_NOT_FOUND</code> and <code>SESSION_NOT_FOUND</code> in favour of <code>USER_UNAUTHORIZED</code></li><li>Removed <code>nativeId</code> from params</li><li><code>Proxy-Authorisation</code> header changed to <code>X-Server-Authorization</code></li></ul>
1.0.1	15.06.2020	<ul style="list-style-type: none"><li>Added Automated Wallet Verifier section</li></ul>
1.0.0	01.06.2020	<ul style="list-style-type: none"><li>First version</li></ul>

## Set up information

*Wallet* owner `SHOULD` provide:

- Endpoints for staging and production environments of the *Wallet*.
- Wallet* name.
- Minimum and maximum bet, maximum exposure (all values in `eur`).

Wallet owner SHOULD be provided with:

- Endpoints for staging and production environments.
- HMAC secret keys for staging and production environments.
- IP of production environment to whitelist.

## Integration

---

Prerequisites:

- Communication SHOULD be done via HTTP/1.1 and encrypted with SSL/TLS .
- All API requests MUST be passed with Content-Type: application/json .
- All API responses MUST be passed with Content-Type: application/json .
- Any additional response or request data, parameters, fields or codes not described in documentation SHOULD NOT be used.
- Character encoding utf-8 SHOULD be set to UTF-8
- All time and date fields SHOULD be in UTC timezone

## Authorization

Authorization MUST be done using the following headers:

- Authorization : Player's authorization with Bearer <token> where <token> comes from /authentication request. It MUST be used to identify a Player.
- X-Server-Authorization : Server's authorization with HMAC scheme using sha256 algorithm and request payload as hex dump. Example of generating the hash: `echo -n $MESSAGE | openssl dgst -sha256 -hmac $SECRET_KEY -hex`

The Operator MUST check and validate headers and return error in case of incorrect authorization data.

## Response codes

All responses MUST return appropriate response's codes.

Code	Status
2XX	Successful
4XX	Application Error
5XX	Server Error

## Errors

All errors MUST follow unified structure.

```
{
  "error": {
    "message": "Token expired",
    "code": "PLAYER_UNAUTHORIZED"
  }
}
```

Field code SHOULD contain one of the following codes:

---

Code	Description
PLAYER_UNAUTHORIZED	Couldn't authorize the player
SERVER_UNAUTHORIZED	Couldn't authorize the server
INSUFFICIENT_FUNDS	Not enough money to make withdrawal
LOSS_LIMIT	Loss limit has been exceeded
UNKNOWN	Unknown error

Field `message` SHOULD contain reason of the error to help troubleshooting. It is not passed to the *Game Client* on production environment, so it can be more descriptive.

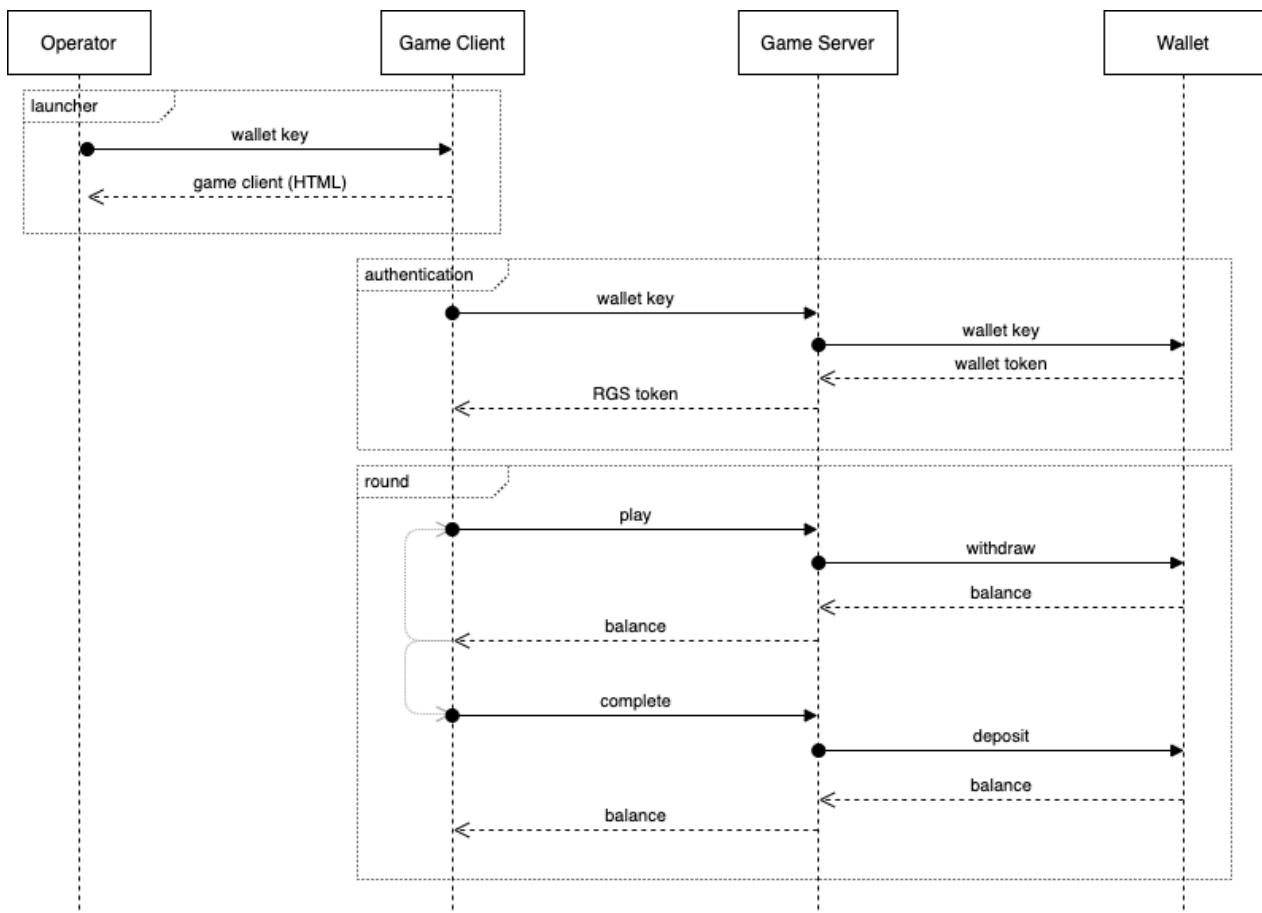
## Idempotence

In case of error, network failure or other reasons selected requests MAY be repeated therefore selected requests MUST be idempotent and return the same outcome.

## Automated Wallet Verifier

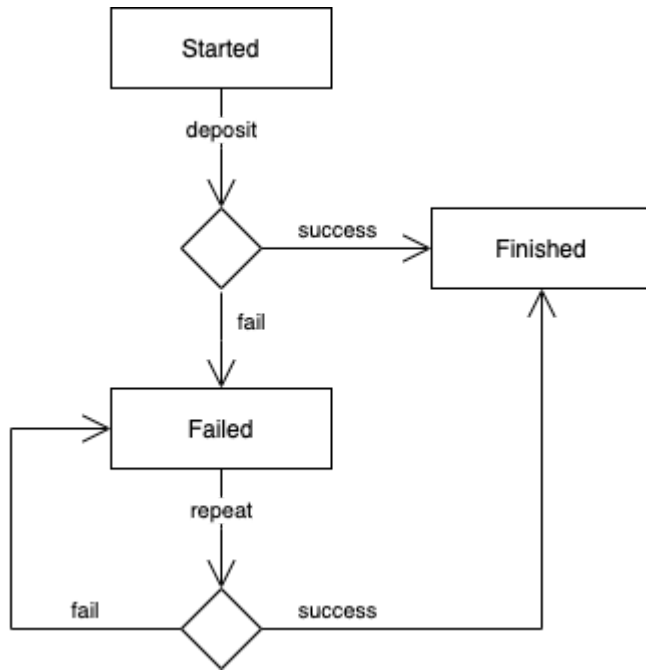
Automated Wallet Verifier offers running automatic test to check integration correctness.

## Flow



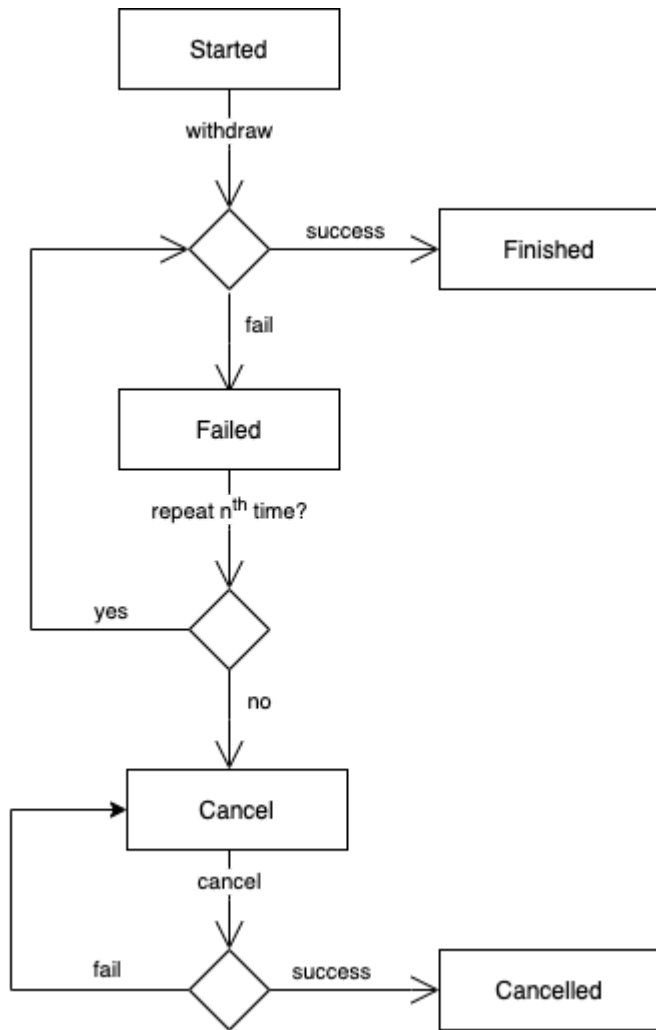
## Deposit flow

In case of failure deposit transactions are repeated indefinitely. First  $n$  repeats MAY come right after the failure and then it still no success then it continues with some time interval until it is successful.



## Withdraw flow

For withdrawals, it MAY be repeated  $n$  times right after the original transaction and if still didn't go through than cancel is initiated. Cancel is repeated with time interval until success.



## Launcher

**Path:** /launch/{fun|real}

**Method:** POST (recommended) or GET

**Authorization:** NO

**Idempotent:** NO

To launch the game the *Operator* should embed `<iframe>` pointing to `<URL>/launch` .

### Parameters

Name	Type	Required	Example	Description
provider	string	REQUIRED	myProvider	Provider Id
wallet	string	REQUIRED	myWallet	Wallet Id.
operator	string	REQUIRED	myOperator	Operator Id
game	string	REQUIRED	myGame	Game Id

Name	Type	Required	Example	Description
key	string	OPTIONAL in fun mode, REQUIRED in real mode	dnsa89me329jdos	-
Operator's_ session initialisation key which SHOULD be active only once after generation and expire after 4 hour				
currency	string	OPTIONAL	eur	Player's currency
language	string	OPTIONAL	en	-
Player's_ language code in ISO 639-1 format				
lobbyUrl	string	OPTIONAL	http://lobby.url	URL to Operator's lobby
depositUrl	string	OPTIONAL	http://deposit.url	URL to Operator's in-game deposit web page

## Response

Returns HTML web page containing the game

**Path:** /launch/replay

**Method:** GET or POST

**Authorization:** NO

**Idempotent:** NO

To launch the game's replay (round summary) the Operator should embed `<iframe>` pointing to `<SERVER_URL>/replay` .

## Parameters

Name	Type	Required	Example	Description
provider	string	REQUIRED	myProvider	Provider Id
game	string	REQUIRED	myGame	Game Id
roundId	string	OPTIONAL	123e4567-e89b-12d3-a456-426614174000	Round Id, one round can have multiple transactions
wallet	string	REQUIRED	myWallet	Wallet Id.
operator	string	REQUIRED	myOperator	Operator Id

## Response

Returns HTML web page containing the game

## API

---

**Path:** /authenticate

**Method:** POST

**Authorization:** X-Server-Authorization

**Idempotent:** NO

Authenticates a *Player* and starts a new session.

### Parameters

Name	Type	Required	Example	Description
operator	string	REQUIRED	MyOperator	Operator Id
key	string	REQUIRED	dnsa89me329jdos	Wallet's_ session initialisation key which SHOULD be active only one time after generation
wallet	string	OPTIONAL	MyWallet	Wallet's name

### Response

Name	Type	Required	Example	Description
nativeId	string	REQUIRED	user123432	_
Wallet's_ Player Id. MUST be unique across entire all operators and brands				
token	string	REQUIRED	db76b22745bca2cd	Authorization token which will be sent in Authorization token in order to verify _
Player_. Preferably it should expire after 7 days				
balance	float	REQUIRED	100.85	Player's current balance in player's currency
currency	string	REQUIRED	eur	Player's currency, <b>MUST NOT change once set!</b>
brand	string	REQUIRED	YellowCasino	Operator's brand
nickname	string	OPTIONAL	JohnnyBravo	Player's nickname

---

Name	Type		Example	Description
gender	string	OPTIONAL	m	Player's gender ( m or f )
country	string	OPTIONAL	uk	–
Player's_ country code in <a href="#">ISO 3166-1 Alpha-2</a> format				
jurisdiction	string	OPTIONAL	mt	–
Player's_ jurisdiction in case of specific market requirements i.e. MT , UK , DK				

**Path:** /balance

**Method:** POST

**Authorization:** X-Server-Authorization and Authorization

**Idempotent:** YES

Fetches *Player's* current balance in *Player's* currency.

#### Parameters

Name	Type		Example	Description
nativeId	string	REQUIRED	user123432	Wallet's Player Id
playerId	string	OPTIONAL	user123432	Adapter's Player Id

#### Response

Name	Type		Example	Description
balance	float	REQUIRED	100.85	Player's current balance in <i>player's</i> currency

**Path:** /transaction

**Method:** PUT (recommended) or POST

**Authorization:** X-Server-Authorization and Authorization

**Idempotent:** YES

Withdraws from or deposits money to *Player's* account. If transaction with given `transactionId` was already processed then don't modify the account and return current balance.

#### Parameters

Name	Type		Example	Description
nativeId	string	REQUIRED	user123432	Wallet's Player Id
playerId	string	OPTIONAL	user123432	Adapter's Player Id



Name	Type		Example	Description
transactionId	string	REQUIRED	321e4567– e89b–45d3– b594– 41234174249	Transaction Id
type	string	REQUIRED	withdraw	withdraw reduces Player's balance. deposit increases _ Player's_ balance.
provider	string	REQUIRED	superProvider	Provider's name
amount	float	REQUIRED	10.56	Amount in _ Player's_ currency to withdraw from the account. Withdrawal of 0 (zero) amount MUST be supported.
jackpotAmount	float	OPTIONAL	0.0017	Part of amount which goes to jackpot contribution (in case of withdrawals) or jackpot win (in case of deposits). It is only for information purpose, this value should not affect _ Player's_ balance on top of 'amount'. Please note contribution can have more than usual 2-decimal places
game	string	OPTIONAL	superSlot	Name of the game
roundId	string	OPTIONAL	123e4567– e89b–12d3– a456– 426614174000	Round Id, one round can have multiple transactions
roundFinished	boolean	OPTIONAL	true	Indicates whether round has finished
category	string	OPTIONAL	promo	Category of transaction: normal, side, tip, promo, jackpot
name	string	OPTIONAL	Royal Match	Name of the transaction
promotionType	string	OPTIONAL	free_round	Type of promotional tool free_round, tournament, mystery (can be more)
promotionId	string	OPTIONAL	myCampaign123	Id of promotional campaign

## Response

---

Name	Type		Example	Description
balance	float	REQUIRED	100.85	Player's current balance

---

**Path:** /cancel

**Method:** DELETE (recommended) or POST

**Authorization:** X-Server-Authorization and Authorization

**Idempotent:** YES

Returns once withdrew money to original account (deposits **MUST NOT** be cancelled). If transaction with given `transactionId` was already canceled then don't modify the account and return current balance. It **MUST NOT** be possible to create new transactions with `transactionId` which was once cancelled.

#### Parameters

Name	Type		Example	Description
nativeId	string	REQUIRED	user123432	Wallet's Player Id
playerId	string	OPTIONAL	user123432	Adapter's Player Id
transactionId	string	REQUIRED	321e4567-e89b-45d3-b594-41234174249	Transaction Id

#### Response

Name	Type		Example	Description
balance	float	REQUIRED	100.85	Player's current balance

---