Casino Floor Live
Operations Manual
For
Operators
And
Aggregators

Casino Floor Live Operations Manual

Change Log

Version 1.13

5.2.2 - Launching a session for player - overide referral domain with domain_id for domain setup in back office.

Version 1.12

6.4.2 GetLimits - Deprecated

3.3 Setting Betting Limits

Version 1.11

Client Control Room is now accessed through the Live Solutions website clients interface. There is no need to establish a separate control room with associated API authentication.

Version 1.10

5.2.2 Control Room initiation - can override the api end point mapping.

6.4.4 CreditUser - enhanced post now includes indication of roundEnd and a breakdown of the constituent bets relating to winning and losing bets.

6.4.5 debitUserFailed - a new post created in the case of a failed debitUser post in order to ensure no transaction anomalies occur because of hardware failure mid post.

Version 1.9

6.4.1 Crypto Milli balances

6.4.2 Crypto Milli playing limits

6.4.3 Crypto Milli Debits

6.4.4 Crypto Milli Credits

Version 1.8

3.1.2 Addition of Domain Groups (bespoke operator player networks)

5.23 Addition of ability to set player language on launch

6.4.5 Deprecation of presenter fan club notifications through api

8.0 Languages and translations

Version 1.7

Removal of licenseeid fixed parameter. If required it should be provided in the initialisation as a custom parameter.

Version 1.6

Added Licensee and Operator Control Room Features

Version 1.5

Unified Lobby with additional filters and searches

Version 1.4

Bets array in both debitUser and creditUser requests now contain an extra field “betName” which gives a descriptor relevant to the “gameType”.

Customisable Lobby for both game choice and presenter choice

Version 1.3

Includes advice on iframe attributes to allow broadcast stream for players, private presenters and private control room staff

Version 1.2

Changed HMAC hash from MD5 to SMA256

Changed from multiple endpoints to single endpoint with parameter indicating the “action” of the request.

Added custom parameters to Launch

Version 1.1

Added array of bets on credit and debit requests

Version 1.0

First Release

Casino Floor Live Operations Manual

Contents

1. Introduction

The Casino Floor Live platform has been created to provide end users with the most immersive experience yet developed for online casino table games. Quite literally, every user has the option to live stream themselves and to see live streaming video from other users.

Naturally this involves a more complex technical infrastructure, but this is shielded from the end user.

This manual is designed to give fully comprehensive instructions, with screen shot illustrations, for integration, administration and operations by software aggregators and casino operators.

Following these instructions will give access to many features that assist operators to comply with their obligations relating to Social Responsibility, AML, Age Verification, and Dispute Resolution and of course it can also provide opportunities for excellent face to face VIP player contact and incentives.

2. Integration

2.1. Integration Concepts

Live Solutions provides you with access to a “Client Control Room” which puts you in control of the whole integration process. You are in control of all of the settings required to set up your own API to function with the “Casino Floor Live” platform.

Live Solutions has developed the platform to have a very simple and efficient integration. You will need to develop and expose at least one Restful API, but you have the flexibility to develop multiple API endpoints if you wish to segregate by domain name.

Our systems will send post requests to your API(s) in order to authenticate users and also in the case of players, to make requests and notify your systems of events and outcomes from the games.

2.1.1. User Privileges

Once you have accessed your client control room with your first user (the Super User), you will be completely in control of granting access to other users and giving them appropriate privileges for their roles.

2.2. Integration Sequence

2.2.1. Client Control Room

You will need to provide login credentials for your first user in the Client's Control Room which can be accessed through the Clients section of the Live Solutions Website. The credentials used are email address and password. If you prefer a password will be created for initial access and you may change the password at any time.

Once you have access to the Client control room you can create new users and give them the privileges they will need to carry out their roles.

You will also have access to all of the settings required to set up the API integration and the branding of the various game interfaces.

2.2.2. Game Launch URLs

The launch URLs you will require are as follows:

Test Environment: -

Players – https://test.casinofloor.livesolutions.im

Production Environment:-

Players – https://casinofloor.livesolutions.im

The IP addresses of servers which will post to both your test and production APIs are 52.51.247.160 and 52.30.244.249. We recommend that you filter out any other IP addresses to prevent unauthorised access

2.2.3. Develop and Expose APIs

The API specifications are contained in section 6 of this manual and you should refer to that section for detailed information.

2.3. API Endpoints

2.3.1. Player APIs

You can specify as many API endpoints for players as you like mapped to different domain names or all domain names mapped to the same API as you like. Player API endpoints require the full specification as described in Section 6 of this manual. You will need to develop and expose each API endpoint before adding it to your API Endpoint register.

2.4. Branding

Each separate site operating through your license can be individually branded. Once you have uploaded both the logo and card back design for the site you can associate them with the domain either on creation or by updating the domain later.

2.4.1. Logos

You may upload as many logo image files as you like. Ideally they should be square in aspect and the file size should not exceed 1Mb. The player interface will do its best to accommodate irregular image dimensions but it is best to checkout how each file looks with each game interface.

2.4.2. Card Designs

You may upload as many card back designs as you like. There is a compulsory image dimension of 175px x 125px.

2.5. Domains

Our systems recognise users by detecting from which domain they have been referred and using the API Endpoint associated with it to authenticate and post betting events and outcomes. It is therefore necessary to register any domain from which a user will be referred. You may register as many domains as you like.

2.6. Testing

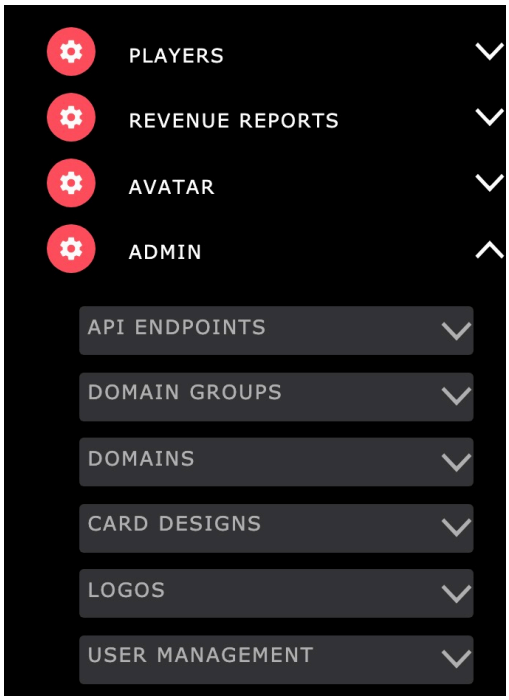
We provide you with a fully functional test environment so that you can perfect your API before going live.

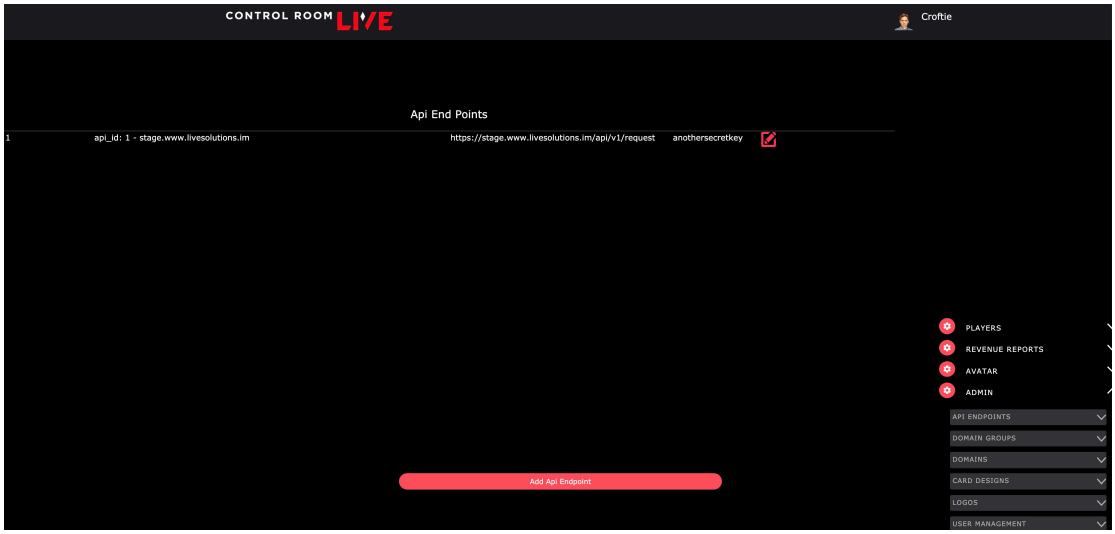
3. Operations

3.1. Administration

Requires “licensee.admin” or any of the other “licensee.admin” sub privileges.

If The Administration menu item cannot be seen then the appropriate privilege needs to be applied to the user by a SuperUser or any user with the “licensee.admin.users” privilege.





3.1.1. API End Points

Requires “licensee.admin” or “licensee.admin.apis” privilege.

API Endpoints are the records which tell our servers where to post requests relating to both your players and your control room users. They are associated to domain records.

Selecting API Endpoints initially displays a list of your API Endpoints already registered.

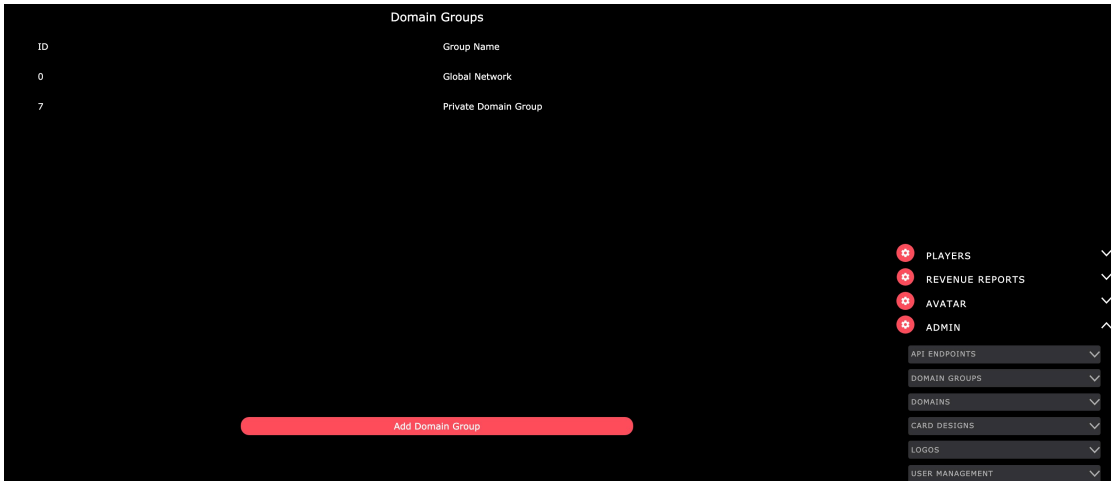
You may create new API Endpoints or update existing ones.

In order to create a new API Endpoint, you will need to specify a name which your users will use to identify the API Endpoint. You will also need to specify the secret phrase used in the HMAC authentication together with the URI used.

This gives our servers all the information it needs to make requests to that API endpoint.

When you create domains you will associate an API Endpoint to that domain.

3.1.2. Domain Groups



Requires “licensee.admin” or “licensee.admin.domainsGroups” privilege.

Domain Groups are records which enable the segregation of players so that they only mix with players from the same Domain Group.

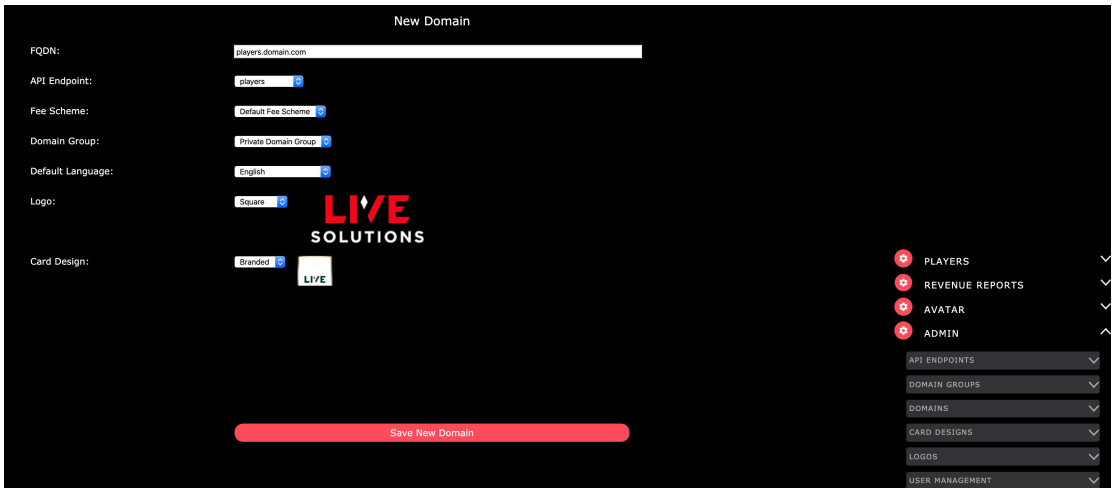
The default Domain Group is the Global Network. Players from domains in this Domain Group will mix at the tables with Players from any other domain which is in the Global Network. Operators have no control over which domains may join this Domain Group. This would be suitable for smaller operators who do not have the volume of business to ensure that players always have company at the tables.

Operators can create their own discrete Domain Groups consisting of one or more domains to segregate their own players.

When you create or update a Domain record you may specify which Domain Group the domain belongs to. Players from this domain will then only mix with other players from this Domain Group.

When you create a Domain Group you should name it in a manner so that it is easy to recognise when creating or changing domain records.

3.1.3. Domains



Requires “licensee.admin” or “licensee.admin.domains” privilege.

Domains are records which you create which our servers use to route api requests, set default languages and branding images.

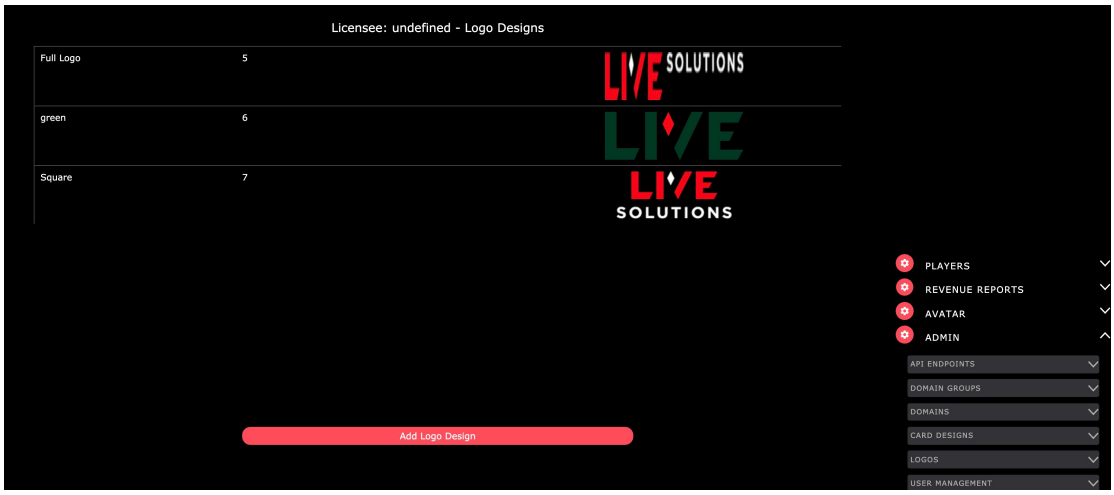
Selecting Domains initially displays a list of your existing domains which also shows the API Endpoint, Fee Scheme, logo, card back design and Domain Group associated with each domain.

You may create new domains or update existing ones.

You will need to have created the API Endpoint, logo, card-back design and Domain Group records already created before creating a new Domain.

There is a default Domain Group named “Global Network” but any other discrete Domain Groups will need to have been created before creating a new Domain.

3.1.4. Logos



Requires “licensee.admin” or “licensee.admin.domains logos” privilege.

Logos are records which you create which tells our servers which branding image to display.

Selecting Logos initially displays a list of your existing logos.

You may create as many new logo records as needed by dragging the image files onto the browser window or clicking on the upload icon and then selecting the file from your local computer.

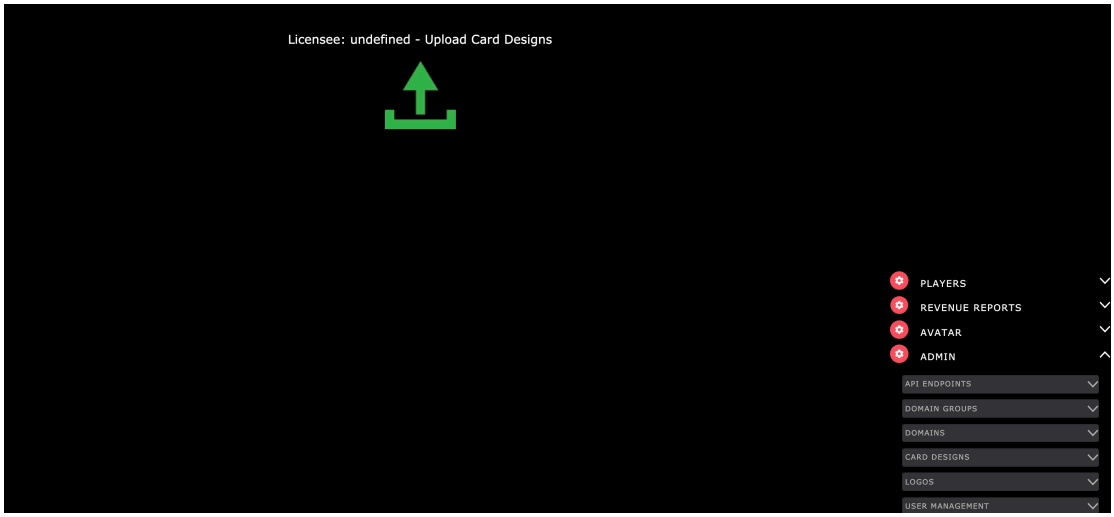
3.1.5. Card Designs

Requires “licensee.admin” or “licensee.admin.domains cards” privilege.

Card Designs are records which you create which tell our servers which branded card back design to display in card games.

Selecting Card Designs initially displays a list of your existing card back designs.

You may create as many new card back designs as needed.



Card back designs must be 175px x 125px.

There is a Photoshop template for card back designs which your account manager can send to you if required.

You may create as many new card-back design records records as needed by dragging the image files onto the browser window or clicking on the upload icon and then selecting the file from your local computer.

3.2. User Management

Requires “licensee.admin” or “licensee.admin.domains users” privilege.

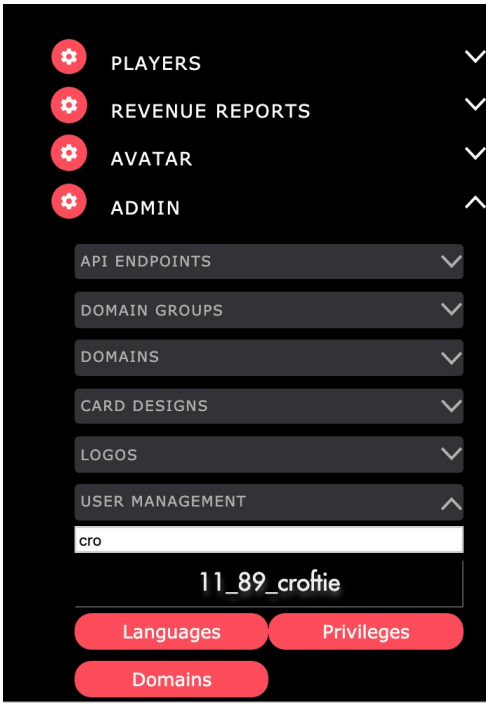
New users are created by any other authorised user using the “add user” button. There is a short form to complete which sets the contact details and credentials and the user type.

This is followed by a consolidated options selection where you can set :-

1. The domains that the user is authorised to operate.
2. The languages that the user has (compulsory for presenters or they will not appear in the appropriate lobby).
3. The privileges that the user has to access the various sections of the control room.

It is also possible to upload and approve a profile photo (compulsory for presenters. If the presenter does not have an approved profile photo they will not appear in any lobby). Profile photos can also be uploaded by the presenter themselves from the presenter interface.

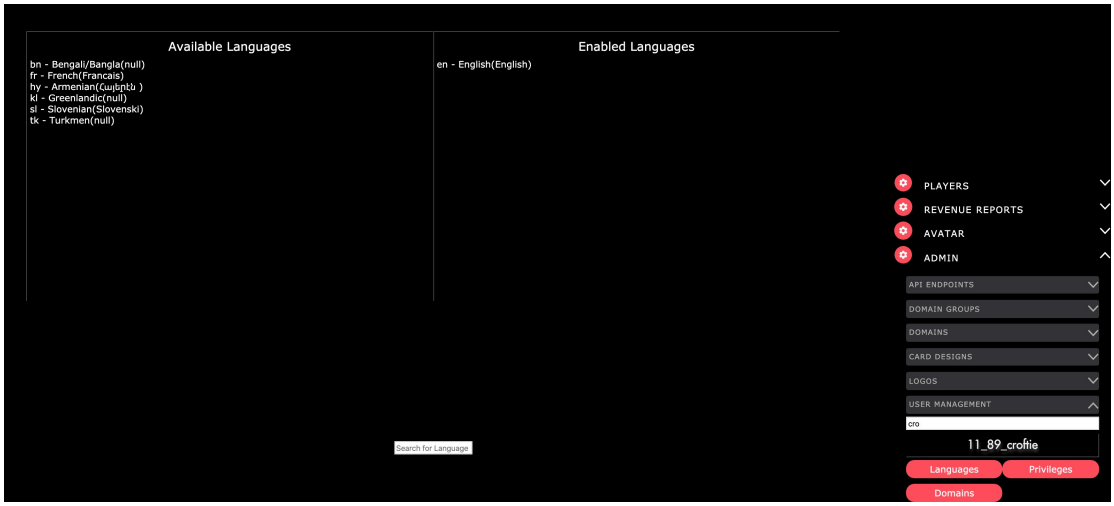
You may search for any user on your system using the search field in the



menu bar.

Selecting the user allows management of the languages, privileges and domains associated with that user.

3.2.1. Languages

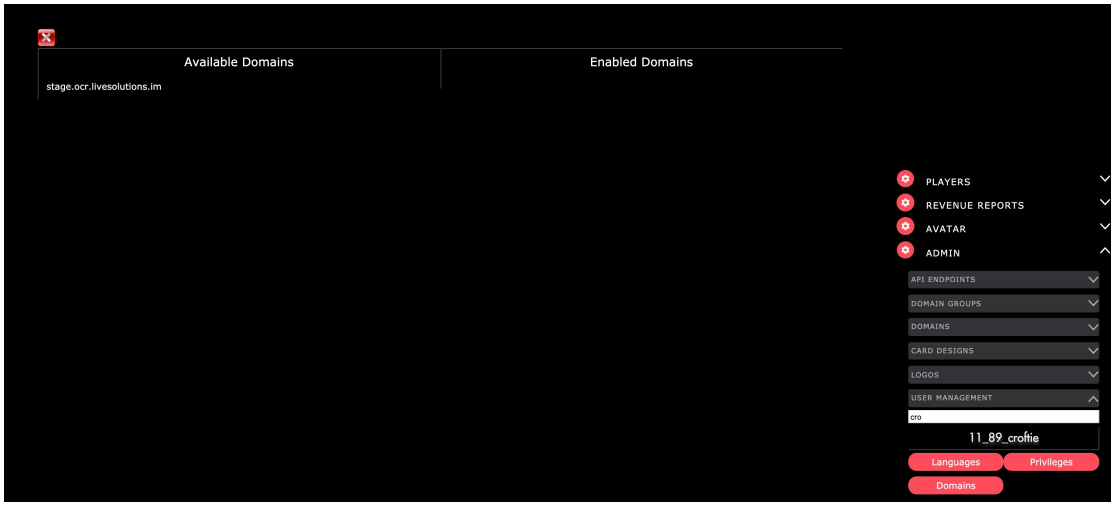


Requires “licensee.admin”, “licensee.admin.users” or “licensee.admin.users.languages” privilege.

The Language Manager can be used to allocate multiple languages to each user.

Languages can be searched for using the search field at the bottom of the page. Languages are added to the users record by clicking on the language in the left hand pane and removed by clicking on the language in the right hand panel.

3.2.2. Domains



Requires “licensee.admin”, “licensee.admin.users” or “licensee.admin.users.domains” privilege.

Superusers have access to and can manage the users of any domain but all other users can only access and manage users and players from the domains for which they are authorised.

Domains are added to a users record by clicking on the domain in the right hand pane and removed from a users record by clicking on a domain in the left hand pane

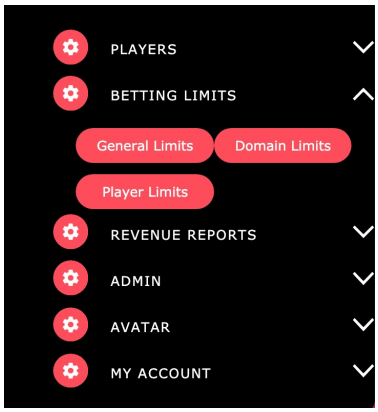
3.2.3. Privileges

Requires “licensee.admin”, “licensee.admin.users” or “licensee.admin.users.privileges” privilege.

Once a user has been selected and privileges selected, clicking on the privilege will add or remove that privilege.

3.3. Setting Betting Limits

Requires the “licensee.bettingLimits” privilege



The Casino Floor Live platform comes with some betting limits for all of the games and which convert from the base currency to all of the currencies supported with logical minimum and maximum limits for each currency.

You can override the defaults with either universal limits for each game or you can set different limits for each game within a particular domain and you can even apply unique betting limits for each game for individual players.

Player limits take priority over domain limits, which take priority over game limits which take priority over defaults.

The Betting Limits Module therefore provides complete flexibility in the setting of the limits to suit all types of operator and level of risk acceptable.

3.3.1. System Defaults

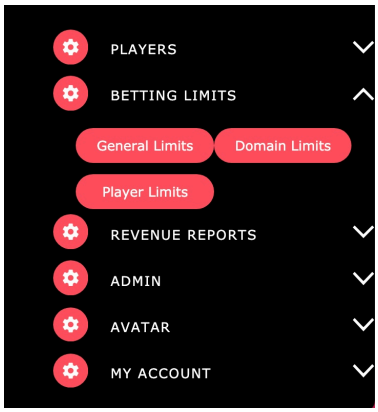
If you do not override the system defaults then the following limits will apply:

Game	Min	Max
Blackjack	€1.00	€5000.00
Roulette	€1.00	€200.00
Baccarat	€1.00	€5000.00
Asian Baccarat	€1.00	€5000.00
Craps	€1.00	€5000.00
Dragon Tiger	€1.00	€5000.00
Fan Tan	€1.00	€5000.00

All betting limits, whether defaults or overrides automatically convert to other currencies so that the limits are logical.

3.3.2. Betting Limit Overrides By Game

Requires the “licensee.bettingLimits.licensee” privilege.



Selecting “General Limits” will display a list of the limits set universally by game. Selecting “Add New Universal Limit” will display a form which will allow the setting of limits for each game. The saving of a limits record for a game automatically removes the existing general limit for that game.

To remove a limit, click on the row.

3.3.3. Betting Limit Overrides by domain

Requires the “licensee.bettingLimits.licensee” privilege.

Selecting “Domain Limits” will display a list of the limits set by game for each domain. Selecting “Add New Domain Limit” will display a form which will allow the setting of limits for each game within a particular domain. The saving of a limits record for a game in a domain automatically removes the existing general limit for that game in that domain.

To remove a domain limit, click on the row.

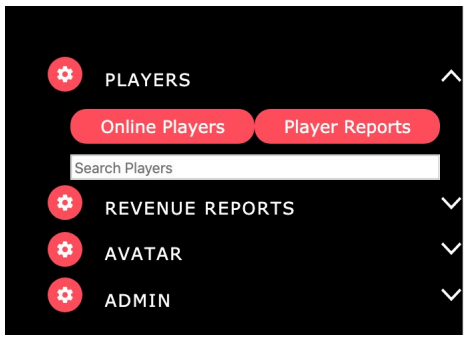
3.3.4. Player Betting Limits

Requires the “licensee.bettingLimits.player” privilege.

Selecting “Player Limits” will display a list of the limits set by game for individual players. Selecting “Add New Player Limit” will display a search field which will display a list of matching players from your domains. Clicking on a particular player displays a form which will allow the setting of limits for each game for that player. The saving of a limits record for a player automatically removes the existing general limit for that game in that player.

3.4. Player Management

Requires “licensee.players” privilege



3.4.1. Online Players

Clicking on the “Online Players” displays a list of all the online users in the domain for which the user is authorised.

3.4.2. Search For Players

Alternatively, it is possible to search for users and this will display users which are both online and offline in the domains for which the user is authorised.

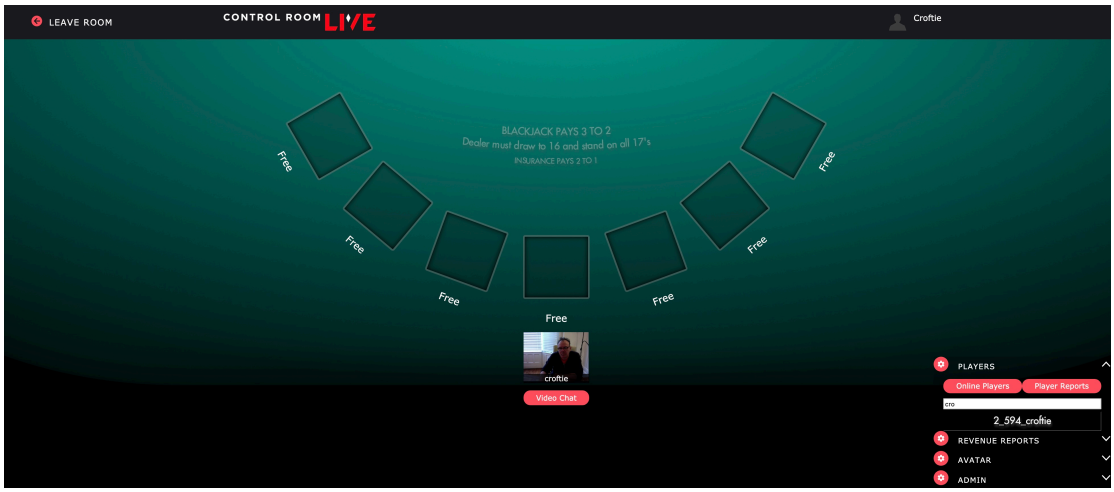
3.4.3. Selecting Players

By clicking on a user new buttons are displayed which relate only to that user.

3.4.4. Player Reports

Clicking on Player Reports displays a list of the recent rounds played and allows the user to view instant replays on any round.

3.4.5. Video Chat with Players



If the player is at a table, a join table button will appear. When a user joins a user at the table he is able to view the action in real time and view the players at the table and the presenter.

Users are only able to interact with players in the domains for which they are authorised. Clicking on the player video displays a new “video chat” button. Clicking on this button opens a private video chat between the player and the user.

Other users at the table no longer see the player’s or the user’s video.

It can therefore be used for any purpose, including promotion.

3.5. Reporting

The following real-time reports can be displayed for any month where there is financial activity.

3.5.1. License Fees

3.5.2. Revenue by Day

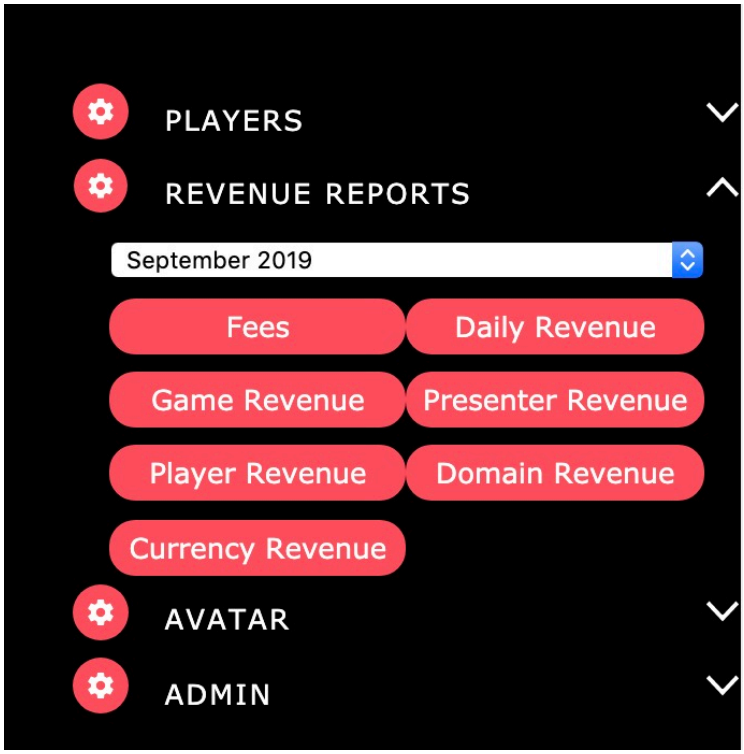
3.5.3. Revenue by Game

3.5.4. Revenue by Presenter

3.5.5. Revenue by Player

3.5.6. Revenue by Domain

3.5.7. Revenue by Currency



3.6. User Avatars

All users have the option to create their own avatar.

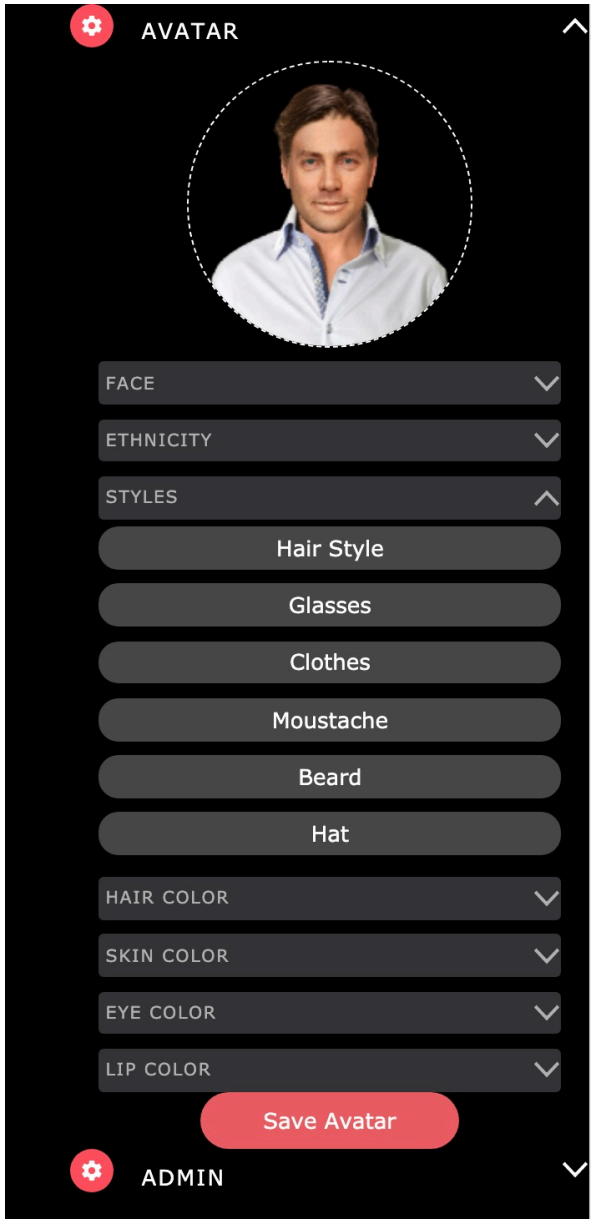
If no avatar has been created then a generic gender neutral silhouette avatar is displayed.

Users can choose gender, ethnicity, hair colour, skin colour, eye colour and lip colour.

There is a whole range of additional options such as hairstyles, glasses and sunglasses, moustache styles, beard styles and lots of different hats and clothes.

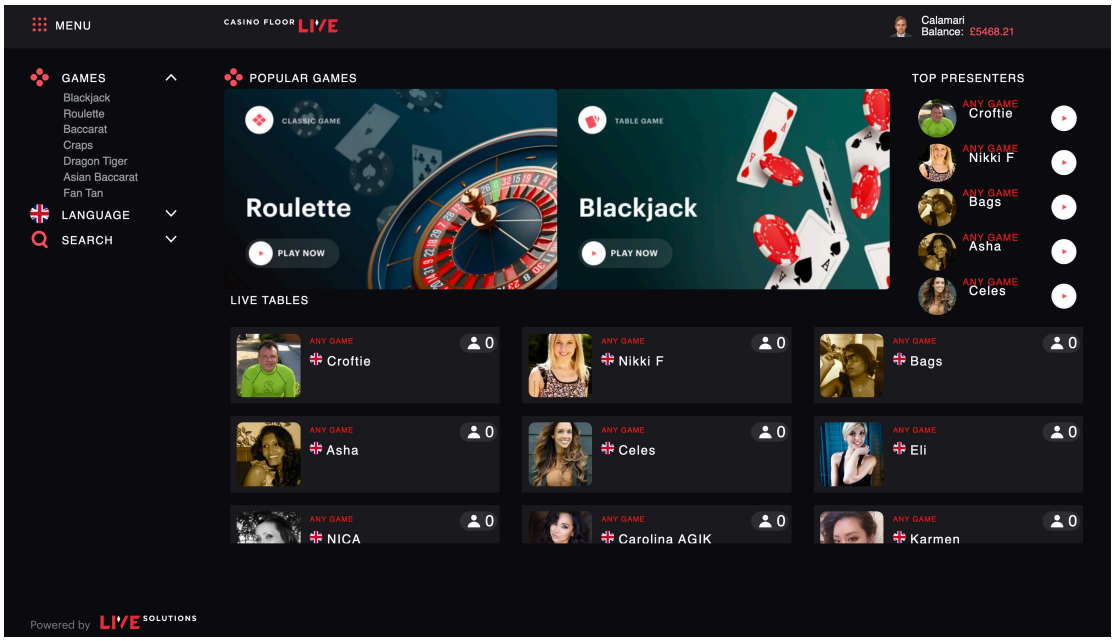
Players also have the option to create avatars which are displayed if they elect not to display video (sound only).

The control room user avatar is only displayed in a very small number of circumstances, but usage gives an understanding of the options available to players and the avatars can be used in email communication with players as well.



4. Player Experience

4.1. Lobby



4.1.1. Header Panel

The header displays the nickname of the player together with their user balance.

4.1.2. Menu Sidebar

This displays the game selection list and as games are added to the platform they will appear in the players lobby automatically.

When a player selects a game, the list of tables available is filtered to show only those tables available for that type of game.

Players can also select language. Selecting a language displays all labels and instructions in that language (with English as a fall back where that language has not been translated) but more importantly it displays tables where the game is being hosted (presented) in that language.

Players can also use the search feature to find their friends or favourite presenters if they are online.

4.1.3. Popular Games

Displays shortcuts to the most popular games.

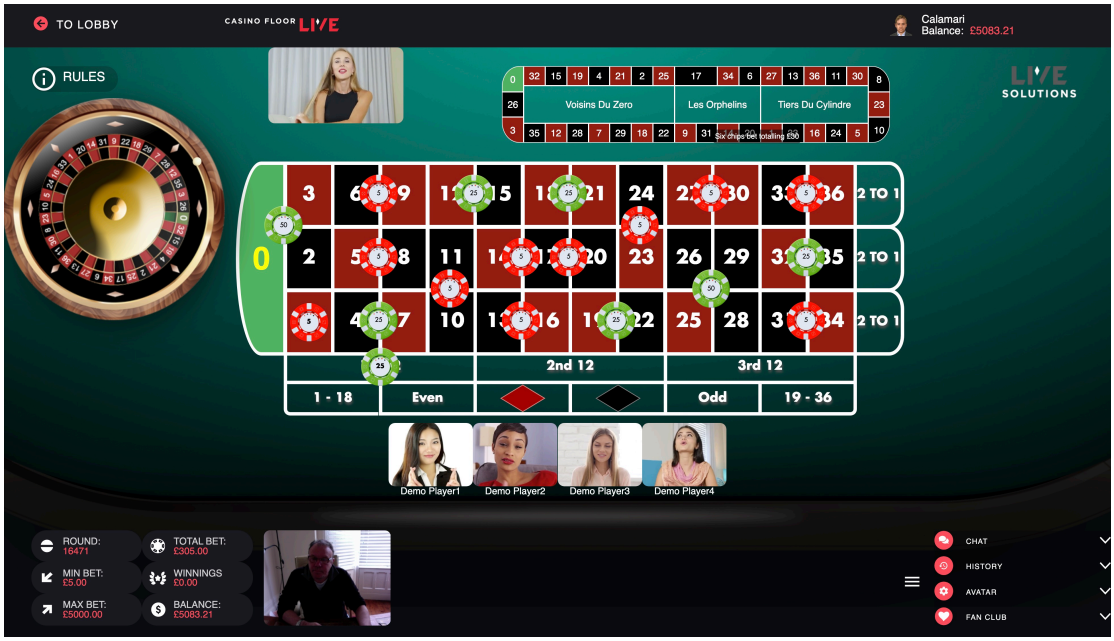
4.1.4. Top Presenters

Displays the most popular presenters.

4.1.5. Table Choice

Displays a list of currently available tables.

4.2. Game Interface



4.2.1. Header Panel

On the left hand side is the Lobby Button, which returns the player to the lobby.

The player's nickname and current balance are displayed on the right.

4.2.2. Game Panel

The "Game Panel" takes the majority of the centre of the viewing panel.

At top left is the "Rules Button" which displays the rules, win table and RTP"

The Table Host or Presenter Video is usually (Except in Roulette) displayed at the top centre. It displays the Presenters nickname and also a favourites button which enables the player to join that presenters fan club.

When one of your players joins a presenter's fan club, when that presenter checks in to host a table, a message is sent by our API to your API Endpoint notifying you. You have the option of informing that player that the presenter has just come online.

Other players' videos are displayed at the bottom. Players have full control over whether other player's videos are displayed by use of the mouse on each individual player video.

When other players bet, their betting chips are initially displayed and then fade. Mousing over the other players' videos reveals that players betting chips and hides all the others.

4.2.3. Control Panel

The control panel displays the round id, total bet for this round, minimum and maximum bet for the game and also any winnings for this round together with the player's current balance.

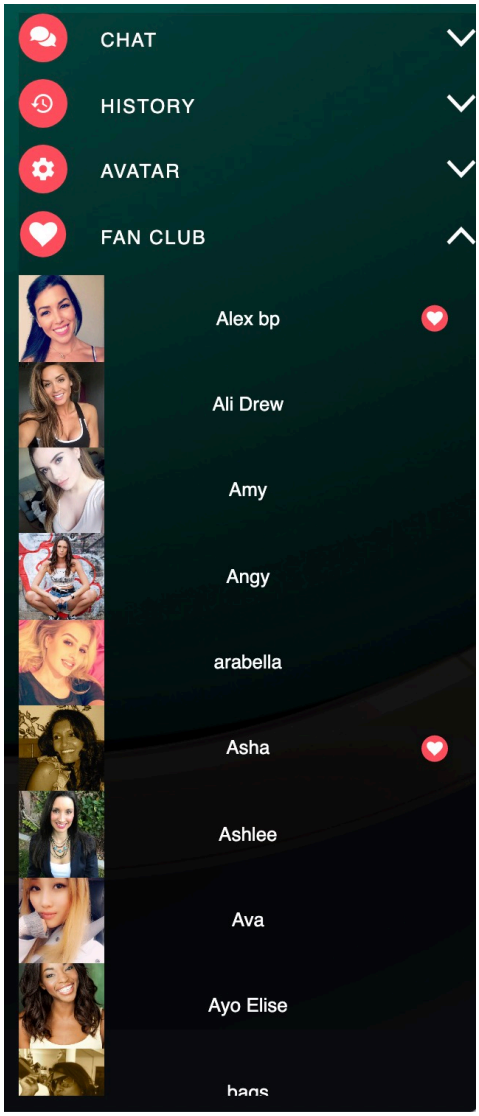
The players own video is also displayed and mousing over reveals the players controls over their own video. They may hide and display at will.

The betting control allows players to select the chip value they wish to bet with and will also display player options required during the game.

4.2.4. Menu

4.2.4.1. Chat

The Chat module allows players to send text messages to the table and to receive text messages from any other player.



4.2.4.2. History

Selection of the History module displays a list of the rounds that the player has played at the current table. It gives details of the Game Type and the total of bets and winnings for that player. By clicking on the link an instant replay of the round will open in a separate tab which will show exactly what happened in any round.

4.2.4.3. Avatar

Players have the option to create their own avatar.

4.2.4.4. Fan Club

Players have the option to join any presenter's fan club either by clicking on the heart icon on the live presenter's video stream or by clicking on the fan club menu and selecting their favourite presenters.

If a player joins a particular presenter's fan club, a web notification is sent to the player every time the presenter checks in to present.

The notification directs the player to the domain which referred the player and is designed as a retention feature.

5. Launching A Client

5.1. Introduction

The Live Solutions platform functions by identifying the domain referring the user and then authenticating that user using the Restful API that has been associated with that domain during its setup.

5.2. Launching A Client Interface

The launch URLs you will require are as follows:

Test Environment: -

Players – https://test.casinofloor.livesolutions.im

Production Environment:-

Players – https://casinofloor.livesolutions.im

To launch a client interface means to include a page in an iframe or to create a new window with the appropriate URL.

5.2.1. Launching into an iframe

When launching into an iframe it is necessary to include the attributes that will allow the user's browser to use both the microphone and camera.

<iframe src = {url} allow="camera; microphone">

5.2.2. Players

Live Solutions does not accommodate "Play For Fun" mode but it does accommodate "guest" mode where the user is not logged in but they can view all of the games and all of the presenters but not launch any of the individual tables.

The platform can either be launched into the Lobby for a particular game, which shows all of the available presenters for that game or can be launched into the General Games Lobby, which shows all the available games.

The platform launches in the language which has been associated with the domain at setup but the player can choose any language from those available, which in itself depends on which presenters are available.

The launch URL for a guest is composed as follows:-

```
LIVE_SOLUTIONS_HOST?guest=true&gameType={gameType}
&language=en
```

The launch URL for real play is composed as follows:-

```
LIVE_SOLUTIONS_HOST?gameType={gameType}&language=en&{custom
key}={custom Value}&{custom key}={custom Value}
```

GET parameters

Parameter	Required	Type	Description
domain_id	No	INT	overide of the referral url to map the api_id and other variables set by domain in the back office
gameType	Yes	STRING	The unique name of the game or “AllGames”
language	No	String	ISO 2 Letter Language Code
{custom keys}	Yes	Custom Values and Types	Custom parameters are passed back to your API during the “initiateSession “ action and allows you to identify your user/licensee

5.2.2.1. Custom Parameters

You can include any other custom parameters. Any Keys ending in “id” will be treated as INT types, any values of either “true” or “false” will be treated as BOOLEAN and all others will be treated as STRING types.

These custom key/value pair parameters will be passed in the “initiateSession” action in order for you to identify your user (and licensee if appropriate).

There are certain reserved words you **cannot** use as custom keys:-

api_id

origin_id

domain_id

gameType

userType

6. API Specifications

6.1. Overview

Live Solutions uses industry best practise and standards. Our specification requires that you develop and expose a Restful API utilising HMAC. We also make available our own Restful API for reconciliation the specification of which can be found in section 7.

6.2. Security

There are two levels of security. You are responsible for:

6.2.1. API Key

Validating every Request based on the HMAC hash sent using the POST method in the Request URL.

6.2.1.1. API Key Overview

The API key is the element of our Restful API client, which provides security, and peace of mind to our clients that any communication received from us is actually from us. The API key is attached to any request as a part of the URI and if checked, provides a means to verify that the request and payload can only have originated from the Live Solutions server.

In short - It stops spoofing and verifies that the content of the payload has not been tampered with.

6.2.1.2. Creation and checking of the API key

You will have provided us with an *API Secret Phrase* at the start of the integration.

The secret phrase is converted into an SHA256 hash.

SHA256("secretphrase") = "EC81D47E5650C9B41215FC39D556B55B20B85E5C1EBCC315F706986A2B389C56"

The SHA256 hash is prepended to the beginning of the JSON payload.

payload = {"gameType":"Blackjack","sessionid":"4lvfvj1gssk2rno34j748l5uv5"}

APIkey=SHA256('EC81D47E5650C9B41215FC39D556B55B20B85E5C1EBCC315F706986A2B389C56{"gameType":"Blackjack","sessionid":"4lvfvj1gssk2rno34j748l5uv5"}')

result = "AEB460EEC5975F48ED470E1200DD987490C8C284779BC8CEEEEAA71C98CACA11"

The API key is attached to the end of the URI for the API endpoint as a Parameter.

http://yourapi.com/your/rest/route/apikey/AEB460EEC5975F48ED470E1200DD987490C8C284779BC8CEEEEAA71C98CACA11

The API key generated by the receiving server should be the same as the one attached to the request URI otherwise the request should be rejected.

6.2.2. IP Whitelisting

You should implement an IP address whitelisting system, which only allows requests that come from the IP addresses of the Live Solutions servers. The IP addresses of servers which will post to both your test and production APIs are 52.51.247.160 and 52.30.244.249.

6.3. Standards

6.3.1. Base Url

`YOUR_HOST/YOUR_ROUTE/apikey/{apikey}`

6.3.2. Request Headers

Content-Type: application/json; charset=utf-8

6.3.3. Request Format

All Live Solutions API Requests contain two parts:

API key ‘hash’ sent using POST method in Request URL , with Hash-based message authentication code (HMAC).

Algorithm used : SHA256

Format: 64 char long hexadecimal string

Raw JSON object sent in body by HTTP POST method containing the Data Parameters.

Each JSON object will contain a variable named “action” which indicates the request type.

6.4. Action Details

6.4.1. Initiate Session

Request Data Parameters

Parameter	Required	Type	Description
action	yes	String	“initiateSession”
{custom parameters}	yes	BOOLEAN, INT or STRING	From Launch Parameters

Example Request

```
{
        "action":"initiateSession",
        "session":"vsts3f3jadgns7ogj81j19gc43"
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
message	yes	INT	
userid	yes	INT	Unique userid will be used in all requests posted in this session
balance	yes	INT	In cents for all currencies. For Crypto currencies which are processed in Millis the balance should be * 100000
currency	yes	STRING	ISO 4217 format

Example Response

```
{
	"error":false,
	"message":"Authorised",
	"userid":123,
	"username":"player1",
	“balance":10099,
	"currency":"GBP"
}
```

Example Error Response

```
{
	"error":true,
	"message":"No Session Found",
}
```

Notes

The balance parameter in the response must be sent as an integer representing the balance in cents. A balance of £100.99 would be expressed as an integer 10099.

A milli Cryptocurrency balance of 1 would be expressed as an integer of 100000 (IE the balance * 1000 * 100 as virtual cents)

6.4.2. Get Limits - Deprecated

Request Data Parameters

Parameter	Required	Type	Description
action	yes	STRING	“getLimits”
userid	yes	INT	From initiateSession response
gameType	yes	String	

Example Request

```
{
	"gameType":"Blackjack",
	"action":"getLimits",
	"userid":123,
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
minBet	yes	INT	In cents
maxBet	yes	INT	In cents

Example Response

```
{
	"error":false,
	"minBet":2500,
	"maxBet":2500000
}
```

Example Error Response

```
{
        "error":true,
        "message":"Player ID Unknown",
}
```

Notes

The gameType parameter in the request must be either a valid gameType string but not “AllGames”

The minBet and maxBet parameters in the response must be sent as integers representing the limits in cents. A limit of £25 would be expressed as the integer 2500.

when a player wishes to use a cryptocurrency denominated in mills, 1 gambling chip = 1/1000. limits should be expressed in mills for the crypto currency.

6.4.3. Debit User

Request Data Parameters

Parameter	Required	Type	Description
gameid	yes	INT	
amount	yes	INT	In cents
rate	yes	DOUBLE	Currency : GBP rate
transid	yes	INT	
action	yes	STRING	“debitUser”
bets amount position timestamp betName	yes yes yes yes no	ARRAY INT INT LONG STRING	Amount in cents and position
userid	yes	INT	From initiateSession response
gbpamount	yes	INT	Total in GBP at rate
roundid	yes	INT	The unique round id

Example Request

```
{
“gameid":1,
“amount":500,
“rate":1.0,
“transid":12475,
“action":"debitUser",
“bets”:[{
    “amount":500,
    “position":7,
    “timestamp":1599992988038,
    "betName":"Ante - seat 3”
  }],
“userid":403,
“gbpamount":500,
“roundid":14440
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
balance	yes	INT	In cents
transid	yes	INT	Your internal transaction reference for our records

Example Response

```
{
"error":false,
"message":"Accepted",
"balance":199698792,
"transid":4947
}
```

Example Error Response

```
{
        "error":true,
        "message":"Insufficient Funds",
}
```

Notes

The gameid parameter in the request must be a valid gameid from the list provided by us as part of integration.

The balance parameter in the response must be sent as an integer representing the balance in cents. A balance of £100.99 would be expressed as an integer 10099.

Any error response will result in the cancelling of all bets, which have not been successfully posted. The player is informed through the “casino floor client”.

Milli based cryptocurrency transactions are expressed in mills with a GBP value conversion at the current rate. 1 milli is worth 1/1000th of the base currency. The milli balance return parameter is expressed as the base balance value * 100000 (the milli balance * 100) as an Integer.

6.4.4. Credit User

Request Data Parameters

Parameter	Required	Type	Description
gameid	yes	INT	
roundEnd	yes	BOOLEAN	Player has no more open bets
amount	yes	INT	In cents
rate	yes	DOUBLE	Currency : GBP rate
transid	yes	INT	
action	yes	STRING	“creditUser”

bets
amount
breakdown
  amount
  transit
  timestamp
closed
position
	yes
	yes
	yes
	yes
	yes
	yes
	yes
	yes
		ARRAY
		INT
		ARRAY
		INT
		INT
		LONG
		BOOLEAN
		INT
			Amounts in cents, position, closed, breakdown array containing amount,transid and timestamp of constituent bets

userid	yes	INT	From initiateSession response
gbpamount	yes	INT	Total in GBP at rate
roundid	yes	INT	The unique round id

Example Request

```
{
“gameid":1,
“roundEnd":true,
“amount":0,
“rate":1.0,
“transid":12476,
“action":"creditUser",
“bets":[{
  “amount":0,
  “breakdown":[{
    “amount":500,
    “transid":12475,
    “timestamp”:1599992988038
   }],
  “closed":true,
  “position”:7
  }],
“userid":403,
“gbpamount":0,
“roundid":14440
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
balance	yes	INT	In cents
transid	yes	INT	Your internal transaction reference for our records

Example Response

```
{
"error":false,
"message":"Accepted",
"balance":199698792,
"transid":4947
}
```

Example Error Response

```
{
        "error":true,
        "message":"Unknown userid",
}
```

Notes

The gameid parameter in the request must be a valid gameid from the list provided by us as part of integration.

All bets where the outcome has been determined are included. bets with credit of zero are losing bets and bets with a credit greater than zero are winning bets with the amount being the total return to the player including their original bets which are listed in the breakdown.

The balance parameter in the response must be sent as an integer representing the balance in cents. A balance of £100.99 would be expressed as an integer 10099.

Cryptocurrency transactions in millis are expressed as the base amount / 1000. 1 milli is worth 1/1000th of the base unit.

The milli balance return parameter is expressed as the integer base balance value * 100000 (the base * 1000 * 100)

A failed creditUser post will be re-attempted every 10 minutes until the post is successful or 10 retries are attempted. After 10 retries the post remains on the database and can be picked up by reconciliation api request.

6.4.5. User Debit Fail

Request Data Parameters

Parameter	Required	Type	Description
gameid	yes	INT	
amount	yes	INT	In cents
rate	yes	DOUBLE	Currency : GBP rate
transid	yes	INT	
action	yes	STRING	“debitUser”
bets amount position timestamp BetName	yes yes yes yes no	ARRAY INT INT LONG STRING	Amount in cents and position
userid	yes	INT	From initiateSession response
gbpamount	yes	INT	Total in GBP at rate
roundid	yes	INT	The unique round id

Example Request

```
{
“gameid":1,
“amount":500,
“rate":1.0,
“transid":12475,
“action":"debitUserFailed",
“bets”:[{
    “amount":500,
    “position":7,
    “timestamp":1599992988038,
    "betName":"Ante - seat 3”
  }],
“userid":403,
“gbpamount":500,
“roundid":14440
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
message	yes	STRING	

Example Response

```
{
"error":false,
"message":"Accepted"
}
```

Example Error Response

```
{
        "error":true,
        “message":"Unknown Action”
}
```

Notes

This post will be created if a debit user action fails for any reason. This is not the same as a debit request which is rejected because of lack of funds for instance. HTTP errors and other outages may occur which will lead to the players’s bets being rejected.

This post is designed to ensure that the debit is not processed (in the case of server failure between post request and response for instance).

The userDebitFailure details match the original debitUser allowing the transaction to be identified and cancelled if it was processed prior to failure.

The debitUserFailed post will be attempted every 10 minutes until the post is successful or 10 retries are attempted. After 10 retries the post remains on the database and can be picked up by reconciliation api request.

7. Reconciliation API

7.1. Overview

We provide a Restful API which utilised the same HMAC authentication as defined in section 6 to enable reconciliation.

It allows aggregators and operators to query the Live Solutions databases in order to obtain details of failed and successful transactions for reconciliation purposes.

7.2. Actions

7.2.1. getFailedTransactions

Request Data Parameters

Parameter	Required	Type	Description
action	yes	String	“getFailedTransactions ”
licensee_id	yes	INT	Provided by Live Solutions
startDate	yes	STRING	Formatted date
endDate	yes	STRING	Formatted date

Example Request

```
{
"action": "getFailedTransactions ",
"licensee_id": 1,
"startDate": "2018-03-26 00:00:01",
"endDate": "2018-07-26 23:59:59"
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
transactions	yes	Array	

Example Response

```
{
"error":false,
"transactions":[
{"id":"7773","licensee_id":"1","userid":"594","gameid":"1","roundid":"10531","remote_id":"
2","currency":"GBP","debit":"0.00","debitGBP":"0.00","credit":"50.00","creditGBP":"50.00"
,"rate":"1","fee_type":"1","remoteTransactionid":null,"timestamp":"2018-04-20 14:03:52"}
]
}
```

Example Error Response

```
{
"error":true,
"message":"Licensee ID Invalid",
}
```

7.2.2. GetSuccessfulTransactions

Request Data Parameters

Parameter	Required	Type	Description
action	yes	STRING	“getTransactions”
licenseeid	yes	INT	provided
startDate	yes	STRING	Formatted date
endDate	yes	STRING	Formatted date

Example Request

```
{
"licensee_id": 1,
"action": "getTransactions",
"startDate": "2018-03-26 00:00:01",
"endDate": "2018-07-26 23:59:59"
}
```

Response Data Parameters

Parameter	Required	Type	Description
error	yes	BOOLEAN	
transactions	yes	Array	In cents

Example Response

```
{
"error":false,
"transactions":[
{"id":"7773","licensee_id":"1","userid":"594","gameid":"1","roundid":"10531","remote_id":"2",
"currency":"GBP","debit":"0.00","debitGBP":"0.00","credit":"50.00","creditGBP":"50.00","rate
":"1","fee_type":"1","remoteTransactionid":999,"timestamp":"2018-04-20 14:03:52"}
]
}
```

Example Error Response

```
{
"error":true,
"message":"Parameter Missing",
}
```

Notes
The startDate and endDate parameters in the request must be in the date format shown in the example

id: Transaction id (LS)

licensee_id: licensee id (LS)

userid: user id (Licensee)

gameid: Game id (Live Solutions)

roundid: Round id (Live Solutions)

origin_id: Site or Operator id

currency: currency of transaction

debit: debit amount in currency

debitGBP: debit amount in GBP

credit: credit amount in currency

creditGBP: credit amount in GBP

rate: exchange rate

fee_type: private or public dealer (1 = public)

remoteTransactionid: Licensee transaction reference (null = unsuccessful post request)

timestamp: timestamp

8. Translations and Languages

8.1. Introduction

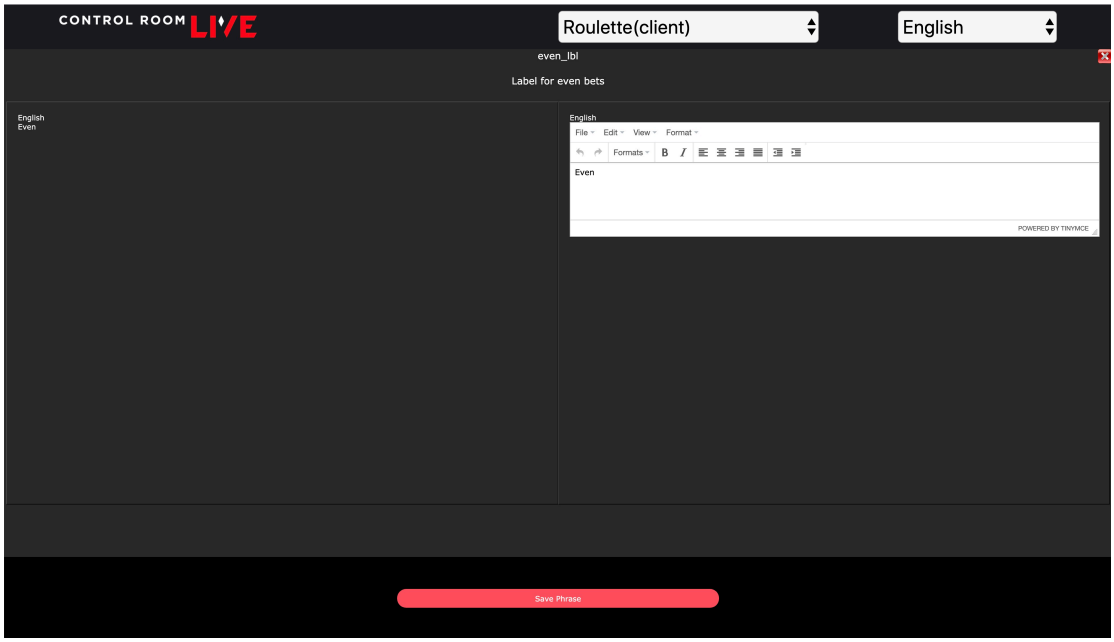
It is important for players to see the lobby and games interfaces in their own language where possible. The translation should be made by a real person (not AI) in order to ensure that it is natural and accurate.

Live Solutions have developed an intuitive Translations and Languages Interface for operators and Aggregators so that they may contribute to the interface labelling and dialog in the most intuitive way

8.2. Concepts

The Live Solutions Translations and Language feature is entirely database driven and can therefore support as many languages as exist within the same platform. We do not hard code any dialog or labels and the content is injected by reference to the player selected language.

8.2.1. Client Side



The client side is the bit which players and control room users see. Each button and label on each interface will need to have a version in the language selected. The Language interface is intuitively grouped by context so for instance if you wanted to provide a label translation for the roulette table in Russian you would select Roulette (client) and Russian in the control bar at the top of the screen.

The interface would then display all the assets which require translations.

The interface highlights assets for which no translation has been provided (in which case the english is displayed in the player interface).

It is possible to view the existing translation and to edit if required.

8.2.2. Server Side

The server side provides translations mainly for the presenters tele-prompt but works exactly the same as the client side.

8.3. Execution

In order to use this feature you will need to request the creation of a user account on our language system for the languages you wish to translate.

Your user may then log in to the languages interface and get to work.