

b2b slots ИНТЕГРАЦИЯ

Оглавление

1. Протокол.....	3
2. Создание связи.....	3
3. Интеграционные API	4
3.1 Игровой URL.....	5
3.2 Auth API.....	6
3.3 Debit API.....	7
3.4 Credit API.....	8
3.5 Rollback API.....	10
3.6 Get Features API.....	11
3.7 Activate Features API.....	13
3.8 End Features API.....	15
4 Словари.....	17
4.1 Словарь валют.....	17
4.2 Коды игр.....	17
4.2 Коды ошибок.....	18
5. Примеры работы API.....	19
5.1 Auth.....	19
5.2 Debit.....	20
5.3 Credit.....	21
5.4 Rollback.....	22
5.5 Get Features.....	23
5.6 Activate Features.....	24
5.7 End Features.....	25

1. Протокол

Интеграция использует Http POST-коммуникации, основанные на типе сообщения Json. Оператор может использовать только HTTPS для связи с нашими серверами.

Для любой синхронной или асинхронной связи в отношении финансовых операций, таких как debit, credit и rollback, бэкэнд Интегратора должен получить ответ с тем же параметром transaction_id, который был отправлен в запросе.

Формат Json используется для связи с бэкэндом для всех видов API.

2. Создание связи

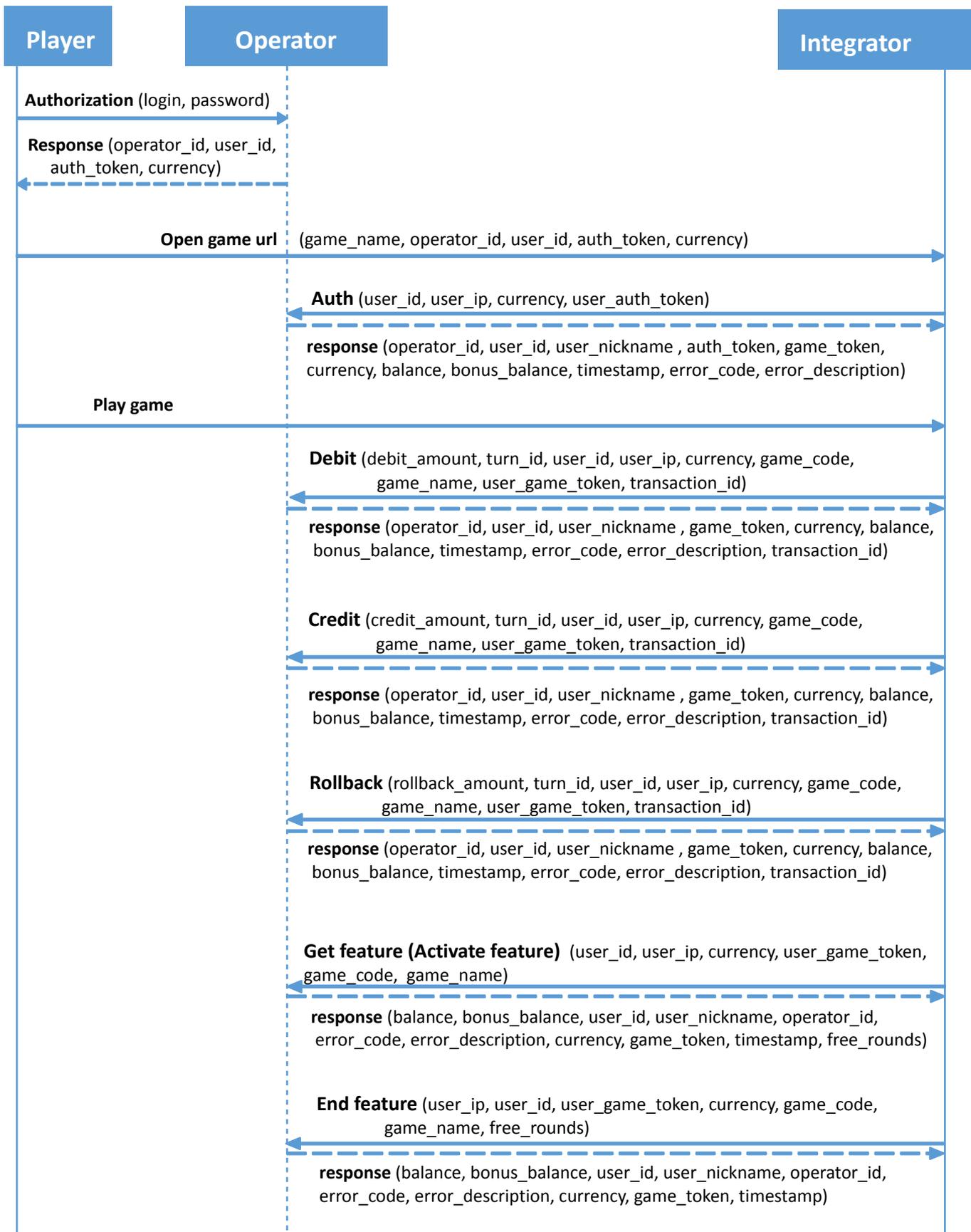
Чтобы создавать backend вызовы, необходимо получить URL-адрес, Тип связи (имя протокола) и точные имена конечных операций.

Ниже приведен список возможных внутренних операций:

- auth
- debit
- credit
- rollback (debit возврат - в случае исключений)
- get features
- end features

3.Интеграционные API

Следующая диаграмма последовательности описывает связь между игроком, Интегратором и оператором, для каждой API.



- Игроки делают авторизацию на стороне оператора и должны получить следующие параметры для использования в URL-адресе игры:

Имя параметра	Тип данных/ Описание
operator_id	int (11)
	Номер оператора, которому принадлежит игрок.
user_id	string (128)
	Уникальный идентификатор для каждого игрока на стороне оператора
auth_token	string (255)
	Уникальный идентификатор, который генерируется и отправляется оператором, Чтобы определить сеанс взаимодействия
currency	string
	Код валюты баланса игрока в определенном сеансе

3.1 Игровой URL

В следующей таблице описываются параметры, предоставляемые оператором для Интегратора, чтобы запустить игру.

```
https://int.apiforb2b.com/games/<game_name>.game?operator_id=<operator_id>
&user_id=<user_id>&auth_token=<auth_token>&currency=<currency>
```

Имя параметра	Тип данных/ Описание
game_name	string
	Уникальное имя игры, которую игрок хочет запустить.
operator_id	int (11)
	Номер оператора, которому принадлежит игрок.
user_id	string (128)
	Уникальный идентификатор для каждого игрока на стороне оператора
auth_token	string (255)
	Уникальный идентификатор, который генерируется и отправляется оператором, Чтобы определить сеанс взаимодействия
currency	string
	Код валюты баланса игрока в конкретной сессии BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)

Пример:

```
https://int.apiforb2b.com/games/BookOfRaClassic.game?
operator_id=0&user_id=1&auth_token=2a9a36b3487129aa899a57b161325ac6e3772174&currency=BTC
```

3.2 Auth API

- **Описание**

Auth API позволяет системе Интегратора проверить информацию, полученной из игрового URL. Только после того, как оператор одобрит эту информацию, игра откроется для игрока.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-auth-user-ingame"
		уникальное название api
data		JsonObject
		Json объект нижеследующих параметров
user_id		string (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_ip		string (30)
		Интернет IP компьютера конечного игрока
user_auth_token		string (255)
		Уникальные идентификатор сгенерированные и отправленные оператором чтобы идентифицировать сеанс
currency		string
		Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
Выходные параметры		
api		String
		уникальное название отправленный Интегратору в запросе, должен иметь значение "do-auth-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
operator_id		Int (11)
		Уникальный идентификатор оператора
user_id		String (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_nickname		String (32)
		Никнейм игрока. Только буквы и числа
balance		String (20)
		Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
bonus_balance		String (20)
		Доступный бонусный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
auth_token		String (255)
		Начальный пользовательский user_auth_token от оператора (путем запуска URL-адреса игры) - мы используем это для проверки.
game_token		String (255)
		Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
error_code		int
		Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, значение кода должно быть равно 0
error_description		String (255)
		Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные

		сведения об ошибке.
	currency	String
		Валюта баланса
	timestamp	String (15)
		Строковое представление времени в формате миллисекунд UNIX.

3.3 Debit API

- **Описание**

Debit API позволяет платформе Интегратора выводить деньги из кошелька игрока через сервер оператора.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-debit-user-ingame"
		уникальное название api
data		JsonObject
		Json объект нижеследующих параметров
user_id		string (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_ip		string (30)
		Интернет IP компьютера конечного игрока
user_game_token		string (255)
		Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
currency		string
		Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
turn_id		BigInt
		Уникальный ключ для идентификации игрового цикла.
transaction_id		String (36)
		Уникальный ключ для конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.
game_code		Int
		Уникальный код игры Интегратора
game_name		String
		Уникальное название игры
debit_amount		String (20)
		Сумма, запрашиваемая для снятия с кошелька игрока. Значение должно быть положительным числом выше нуля. Строковое представление BigDecimal(15,5)
Выходные параметры		
api		String
		Уникальное имя api, отправленное в запросе Интегратору, значение должно быть установлено равным "do-debit-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
operator_id		Int (11)
		Уникальный идентификатор оператора
transaction_id		String (36)
		Тот же transaction_id, который был отправлен в запросе. Уникальный ключ для указания конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.

	user_id	String (128) Уникальный идентификатор для каждого игрока на стороне оператора
	user_nickname	String (32) Никнейм игрока. Только буквы и числа
	balance	String (20) Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
	bonus_balance	String (20) Доступный бонусный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
	bonus_amount	String (20) Сумма бонуса часть от общей суммы debit. *Не может быть больше суммы debit. Строковое представление BigDecimal(15,5)
	game_token	String (255) Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
	error_code	int Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, значение должно быть равно 0
	error_description	String (255) Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
	currency	String Валюта баланса
	timestamp	String (15) Строковое представление времени в формате миллисекунд UNIX.

3.4 Credit API

- **Описание**

Credit API позволяет платформе Интегратора выполнять финансовую операцию по возврату денег в кошелек игрока через сервер оператора. При расчете результатов игрок, который проиграл, имеет credit amount равным 0. Выигравшему игроку, начисляется сумма выплаты.

Имя параметра	Тип данных/ Описание
Входные параметры	
api	String, value = "do-credit-user-ingame" уникальное название api
data	JsonObject Json объект нижеследующих параметров
	user_id string (128) Уникальный идентификатор для каждого игрока на стороне оператора
	user_ip string (30) Интернет IP компьютера конечного игрока
	user_game_token string (255) Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
	currency string Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)

	turn_id	BigInt Уникальный ключ для идентификации игрового цикла.
	transaction_id	String (36) Уникальный ключ для конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.
	game_code	Int Уникальный код игры Интегратора
	game_name	String Уникальное название игры
	credit_amount	String(20) Сумма, запрашиваемая для пополнения кошелька игрока. Значение должно быть положительным числом выше нуля. Строковое представление BigDecimal(15,5)
Выходные параметры		
	api	String уникальное название api, отправленный Интегратору, значение должно быть установлено равным "do-credit-user-ingame"
	success	Boolean В результате запроса api значение всегда должно быть true
	answer	JsonObject нижеследующих параметров
	operator_id	Int (11) Уникальный идентификатор оператора
	transaction_id	String (36) Тот же transaction_id, который был отправлен в запросе. Уникальный ключ для указания конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.
	user_id	String (128) Уникальный идентификатор для каждого игрока на стороне оператора
	user_nickname	String (32) Никнейм игрока. Только буквы и числа
	balance	String (20) Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
	bonus_balance	String (20) Доступный бонусный баланс в кошельке игрока на момент запроса. Значение будет представлено в валюте текущего сеанса. Строковое представление BigDecimal(15,5)
	bonus_amount	String(20) Сумма бонуса от общей суммы credit_amount. *Не может быть больше суммы credit_amount. Строковое представление BigDecimal(15,5)
	game_token	String (255) Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
	error_code	int Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, это значение кода равно 0
	error_description	String (255) Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
	currency	String Валюта баланса
	timestamp	String (15) Строковое представление времени в формате миллисекунд UNIX.

3.5 Rollback API

- **Описание**

Во время транзакции с debit могут быть случаи, когда транзакция должна быть отменена и, следовательно, средства возвращены. Rollback API разработан для работы с исключениями, полученными от оператора (то есть при отключении сервера, либо со стороны оператора возвращается недопустимый ответ об ошибке). Debit транзакция конвертируется в транзакцию rollback с использованием того же идентификатора транзакции transaction_id, а поле суммы debit_amount переименовывается в rollback_amount. Очень важно, чтобы откат был отдельным методом от debit, чтобы избежать ошибок и смешения данных.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-rollback-user-ingame"
		уникальное название api
data		JsonObject
		Json объект нижеследующих параметров
user_id		string (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_ip		string (30)
		Интернет IP компьютера конечного игрока
user_game_token		string (255)
		Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
currency		string
		Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
turn_id		BigInt
		Уникальный ключ для идентификации игрового цикла.
transaction_id		String (36)
		Уникальный ключ для конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.
game_code		Int
		Уникальный код игры Интегратора
game_name		String
		Уникальное название игры
rollback_amount		String (20)
		Сумма, которую необходимо вернуть в кошелек игрока. Значение должно быть положительным числом выше нуля. Строковое представление BigDecimal(15,5)
Выходные параметры		
api		String
		уникальное название api, отправленный Интегратору, значение должно быть установлено равным "do-rollback-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
operator_id		Int (11)
		Уникальный идентификатор оператора

	transaction_id	String (36) Тот же transaction_id, который был отправлен в запросе. Уникальный ключ для указания конкретной финансовой операции. Этот ключ гарантирует, что транзакция будет обработана только один раз.
	user_id	String (128) Уникальный идентификатор для каждого игрока на стороне оператора
	user_nickname	String (32) Никнейм игрока. Только буквы и числа
	balance	String (20) Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
	bonus_balance	String (20) Доступный бонусный баланс в кошельке игрока на момент запроса. Значение будет представлено в валюте текущего сеанса. Строковое представление BigDecimal(15,5)
	game_token	String (255) Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
	error_code	int Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, это значение кода равно 0
	error_description	String (255) Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
	currency	String Валюта баланса
	timestamp	String (15) Строковое представление времени в формате миллисекунд UNIX.

3.6 Get Features API

- **Описание**

Get Features API позволяет платформе Интегратора запрашивать любую бонусную программу, такую как бесплатные прокруты со стороны оператора.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-get-features-user-ingame" уникальное название api
data		JsonObject Json объект нижеследующих параметров
	user_id	string (128) Уникальный идентификатор для каждого игрока на стороне оператора
	user_ip	string (30) Интернет IP компьютера конечного игрока
	user_game_token	string (255) Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
	currency	string Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
	game_code	Int Уникальный код игры Интегратора

Выходные параметры		
api		String
		уникальное название api, отправленный Интегратору, значение должно быть установлено равным "do-get-features-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
	operator_id	Int (11)
		Уникальный идентификатор оператора
	user_id	String (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
	user_nickname	String (32)
		Никнейм игрока. Только буквы и числа
	balance	String (20)
		Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
	bonus_balance	String (20)
		Доступный бонусный баланс в кошельке игрока на момент запроса. Значение будет представлено в валюте текущего сеанса. Строковое представление BigDecimal(15,5)
	game_token	String (255)
		Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
	error_code	int
		Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, это значение кода равно 0
	error_description	String (255)
		Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
	currency	String
		Валюта баланса
	timestamp	String (15)
		Строковое представление времени в формате миллисекунд UNIX.
	free_rounds	JsonObject
		Json Object нижеследующих параметров
	id	int
		Уникальный идентификатор бонуса
	count	int
		Количество бесплатных прокрутов
	bet	int
		Ставка на которой будут запущены прокруты
	lines	int
		Количество линий на которых будут запущены прокруты
	mpl	int
		Мультипликатор бонусна
	cp	double
		Деноминация

3.7 Activate Features API

- **Описание**

Activate Features API позволяет платформе Интегратора оповещать оператора о бонусах, активированных игроком.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-activate-features-user-ingame"
		The unique api name
data		JsonObject
		Json объект нижеследующих параметров
user_id		string (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_ip		string (30)
		Интернет IP компьютера конечного игрока
user_game_token		string (255)
		Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
currency		string
		Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
game_code		Int
		A unique game code in the Integrator
free_rounds		JsonObject
		Json Object нижеследующих параметров
id		int
		Уникальный идентификатор бонуса
Выходные параметры		
api		String
		Уникальное имя api, отправленное в запросе Интегратору, значение должно быть установлено равным "do-activate-features-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
	operator_id	Int (11) Уникальный идентификатор оператора
user_id		String (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_nickname		String (32)
		Никнейм игрока. Только буквы и числа
balance		String (20)
		Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
bonus_balance		String (20)
		Доступный бонусный баланс в кошельке игрока на момент запроса. Значение будет представлено в валюте текущего сеанса. Строковое представление BigDecimal(15,5)
game_token		String (255)
		Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.

	error_code	int
		Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, значение должно быть равно 0
	error_description	String (255)
		Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
	currency	String
		Валюта баланса
	timestamp	String (15)
		Строковое представление времени в формате миллисекунд UNIX.

3.8 End Features API

- **Описание**

End Features API позволяет платформе Интегратора оповещать об окончании предыдущих активированных бонусов.

Имя параметра		Тип данных/ Описание
Входные параметры		
api		String, value = "do-end-features-user-ingame"
		The unique api name
data		JsonObject
		Json объект нижеследующих параметров
user_id		string (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_ip		string (30)
		Интернет IP компьютера конечного игрока
user_game_token		string (255)
		Уникальный идентификатор, который генерируется (в auth api) оператором для определения игровой сессии.
currency		string
		Код валюты баланса игрока в конкретной сессии: BTC, LTC, USD, EUR, RUB, KZT, UAH (может быть расширен в будущем)
game_code		Int
		A unique game code in the Integrator
free_rounds		JsonObject
		Json Object нижеследующих параметров
id		int
		Уникальный идентификатор бонуса
win		double
		Сумма выигрыша в бонусе. Сумма, пополнения кошелька игрока. Значение должно быть положительным числом выше или равным нулю
Выходные параметры		
api		String
		Уникальное имя api, отправленное в запросе Интегратору, значение должно быть установлено равным "do-end-features-user-ingame"
success		Boolean
		В результате запроса api значение всегда должно быть true
answer		JsonObject нижеследующих параметров
	operator_id	Int (11) Уникальный идентификатор оператора
user_id		String (128)
		Уникальный идентификатор для каждого игрока на стороне оператора
user_nickname		String (32)
		Никнейм игрока. Только буквы и числа
balance		String (20)
		Доступный баланс игрока на момент запроса. Значение должно быть представлено в валюте текущей сессии. Строковое представление BigDecimal(15,5)
bonus_balance		String (20)
		Доступный бонусный баланс в кошельке игрока на момент запроса. Значение будет представлено в валюте текущего сеанса. Строковое представление BigDecimal(15,5)

game_token	String (255)
	Не путать с вводом auth_token. Этот токен является токеном сеанса. Уникальный идентификатор, который генерируется оператором для игровой сессии.
error_code	int
	Определяет, был ли запрос успешно обработан. Если ошибка не обнаружена, значение должно быть равно 0
error_description	String (255)
	Строка, описывающая ответ. Эта строка не является сообщением игроку, а содержит подробные сведения об ошибке.
currency	String
	Валюта баланса
timestamp	String (15)
	Строковое представление времени в формате миллисекунд UNIX.

4 Словари

4.1 Словарь валют

Код валюты	Название валюты
BTC	Bitcoin криптовалюта
LTC	Litecoin криптовалюта
USD	Американский доллар
EUR	Евро
RUB	Российский рубль
KZT	Казахстанский тенге
UAH	Украинская гривна
*	*

* Могут быть добавлены любые валюты

4.2 Коды игр

Полный список игр можно получить в формате json выполнив запрос GET методом, указав operator id в параметре operator_id:

```
https://int.apiforb2b.com/frontendsrv/apihandler.api?cmd={%22api%22:%22ls-games-by-operator-id-get%22,%22operator_id%22:%220%22}
```

Получение иконки игры из JSON:

```
{
  "ico_baseurl": "/game/icons/",
  "groups": [
    {
      "games": [
        {
          ...
          "icons": [
            {
              "ic_h": 221,
              "ic_w": 300,
              "ic_name": "ReelKing_300_221.jpg"
            },
            ...
          ],
          "gm_title": "Reel King"
        },
        ...
      ],
      ...
    }
  ]
}
```

```
https://int.apiforb2b.com[ico_baseurl][ic_name]
```

Пример:

```
https://int.apiforb2b.com/game/icons/ReelKing_300_221.jpg
```

* Новые игры всегда в разработке. Данный список может быть расширен в любое время.

4.2 Коды ошибок

Код ошибки	Описание ошибки
0	Завершено успешно
1	Общая ошибка
2	*
3	Недостаточно средств
4	Token не найден
5	Пользователь не найден
6	Пользователь заблокирован
7	Транзакция не найдена
8	Транзакция истекла
9	*
10	*
11	*

* - Зарезервированные коды для будущего использования

Примечание: В том случае, если один и тот же `transaction_id` отправлен более одного раза, Вы должны отправить нам тот же ответ с исходным кодом ошибки. В `error_description` написать `transaction has already proceeded`.

5. Примеры работы API

5.1 Auth

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_auth_token": "638b779ea073609b28374364bf1eb0de488fb9b5",
    "currency": "BTC"
  },
  "api": "do-auth-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "user_id": "1",
    "user_nickname": "Anonymous",
    "balance": "855.67713",
    "bonus_balance": "0.00",
    "auth_token": "638b779ea073609b28374364bf1eb0de488fb9b5",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "timestamp": "1478008017191"
  },
  "success": true,
  "api": "do-auth-user-ingame"
}
```

5.2 Debit

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "turn_id": 4,
    "transaction_id": "2fa7bfb2-a03a-11e6-8347-00ffa41930b3",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
    "debit_amount": "0.02"
  },
  "api": "do-debit-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "transaction_id": "2fa7bfb2-a03a-11e6-8347-00ffa41930b3",
    "user_id": "1",
    "user_nickname": "Anonymous",
    "balance": "855.67713",
    "bonus_balance": "0.0",
    "bonus_amount": "0.0",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "timestamp": "1478008244952"
  },
  "success": true,
  "api": "do-debit-user-ingame"
}
```

5.3 Credit

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "turn_id": 5,
    "transaction_id": "2fa7bfb2-a03a-11e6-8347-00ffa41930b3",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
    "credit_amount": "0.02"
  },
  "api": "do-credit-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "transaction_id": "2fb1fd48-a03a-11e6-8347-00ffa41930b3",
    "user_id": "1",
    "user_nickname": "Anonymous",
    "balance": "855.67913",
    "bonus_balance": "0.00",
    "bonus_amount": "0.00",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "timestamp": "1478008244987"
  },
  "success": true,
  "api": "do-credit-user-ingame"
}
```

5.4 Rollback

Интегратор запрос

```
{
  "data": {
    "user_ip": "127.0.0.1",
    "user_id": "1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "turn_id": 5,
    "transaction_id": "2fa7bfb2-a03a-11e6-8347-00ffa41930b3",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
    "rollback_amount": "0.02"
  },
  "api": "do-rollback-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "transaction_id": "2fb1fd48-a03a-11e6-8347-00ffa41930b3",
    "user_id": "1",
    "user_nickname": "Anonymous",
    "balance": "855.67913",
    "bonus_balance": "0.00",
    "bonus_amount": "0.00",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "timestamp": "1478008244987"
  },
  "success": true,
  "api": "do-rollback-user-ingame"
}
```

5.5 Get Features

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
  },
  "api": "do-get-features-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "user_id": "1",
    "user_nickname": "Anonimous",
    "balance": "855.67913",
    "bonus_balance": "0.00",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "timestamp": "1478008244987",
    "free_rounds": {
      "id": 1,
      "count": 15,
      "bet": 5,
      "lines": 10,
      "mpl": 2,
      "cp": "1.00"
    }
  },
  "success": true,
  "api": "do-get-features-user-ingame"
}
```

5.6 Activate Features

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
    "free_rounds": {
      "id": 1
    }
  },
  "api": "do-activate-features-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "user_id": "1",
    "user_nickname": "Anonimous",
    "balance": "855.6913",
    "bonus_balance": "0.00",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "timestamp": "1478008244987"
  },
  "success": true,
  "api": "do-activate-features-user-ingame"
}
```

5.7 End Features

Интегратор запрос

```
{
  "data": {
    "user_id": "1",
    "user_ip": "127.0.0.1",
    "user_game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "currency": "BTC",
    "game_code": 702,
    "game_name": "BookOfRaClassic",
    "free_rounds": {
      "id": 1,
      "win": 1000
    }
  },
  "api": "do-end-features-user-ingame"
}
```

Ответ Оператора

```
{
  "answer": {
    "operator_id": 0,
    "user_id": "1",
    "user_nickname": "Anonimous",
    "balance": "855.6913",
    "bonus_balance": "0.00",
    "error_code": 0,
    "error_description": "ok",
    "currency": "BTC",
    "game_token": "c3720505ad4dfd2b34cdb55bc524f3ffa6941ad7",
    "timestamp": "1478008244987"
  },
  "success": true,
  "api": "do-end-features-user-ingame"
}
```