

LuckyFit

Интеграция с внешними системами

Основные принципы интеграции

LuckyFit предоставляет API для интеграции с внешними системами на основе http REST запросов. Базовый url для запросов: <https://my.lucky.fitness>. Названия методов в дальнейшем в данной документации будут представлены по отношению к базовому url. Параметры передаются в json. В заголовке каждого запроса необходимо передавать API-KEY, который используется для авторизации.

Структура запроса

Параметры и данные запроса передаются в JSON.

В заголовке запроса необходимо передать параметр Api-Key со значением токена доступа, назначенного вашей организации.

В POST, PUT, DELETE запросах JSON данных передается в теле запроса.

В GET-запросах параметры передаются в переменной params, которая содержит Url-энкодированный JSON-передаваемых параметров.

Например, чтобы получить список счетов, начиная с 01 ноября 2017 года, необходимо отправить такой запрос:

```
https://my.lucky.fitness.com/orders?  
params=%7B%22lo%23closed%22%3A%222017-11-01T00%3A00%3A00%22%7D
```

Структура ответа на запросы

Ответ на запросы передается в виде json со структурой:

success - признак успешности (true,false)

error - строка ошибки в случае success == false

session_expired (true,false) - признак устаревшей сессии, необходима повторная авторизация. Происходит, когда истекло время жизни сессии (более 8 часов бездействия).

data - возвращаемые данные. В зависимости от запроса, может быть простой json-объект, либо массив объектов

total - количество записей в массиве data

Получение классификатора товаров и услуг

путь

`/good_classes`

метод	GET
параметры	может использоваться любая комбинация параметров: good_class_id - идентификатор класса для получения дочерних классов. Если не передан, то возвращаются корневые классы
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей Возвращаемые данные: id - идентификатор класса товара organization_id - идентификатор организации name - название position - номер позиции классификатора good_class_id - ссылка на родительский класс created_at - дата и время создания updated_at - дата и время изменения resource_id - идентификатор изображения annuled - признак аннулирования firm_id - идентификатор внутреннего контрагента группы товаров

Получение списка товаров и услуг

путь	/goods
метод	GET
параметры	может использоваться любая комбинация параметров: good_kind_id - тип товара good_class_id - идентификатор группы абонементов article - артикул
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей Возвращаемые данные: id - идентификатор товара name - название price - цена article - артикул measure_name - единица измерения description - описание good_kind_id - тип товара (0-сырье 1-товар 2-услуга 3-калькуляция) good_class_id - идентификатор класса organization_id - идентификатор организации measure_id - ключ id измерения annuled - признак аннулирования change_price - возможность изменения цены товара при продаже divisible - признак делимости товара tax_id - id налоговой группы barcode - штрих-код supplier_article_id - артикул поставщика товара resource_id - id изображения

Получение классификатора абонементов

путь	/ticket_classes
метод	GET
параметры	может использоваться любая комбинация параметров: ticket_class_id - идентификатор класса для получения дочерних классов. Если не передан, то возвращаются корневые классы
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей Возвращаемые данные: id - идентификатор класса абонемента organization_id - идентификатор организации name - название position - номер позиции классификатора ticket_class_id - ссылка на родительский класс created_at - дата и время создания updated_at - дата и время изменения resource_id - идентификатор изображения annuled - признак аннулирования

Получение списка абонементов из ассортимента

путь	/tickets
метод	GET
параметры	Для получения только абонементов из ассортимента без абонементов клиентов, необходимо передать параметр ticket_id:null может использоваться любая комбинация параметров: ticket_kind_id - тип абонемента (1-клубное членство, 2-пакет тренировок) ticket_class_id - идентификатор группы абонементов

ответ

json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Возвращаемые данные:

id - идентификатор абонеента
ticket_class_id - идентификатор класса
ticket_kind - тип абонеента (0-пробный 1-клубное членство 2-дополнительный)
name - название
description - описание
duration - длительность
time_kind_id - тип длительности (1-мин 2-час 3-сутки 4-мес 5-год)
price - цена
membership_summ - сумма членского взноса
freeze_count - количество заморозок
freeze_days - количество дней заморозок
created_at - момент создания
updated_at - момент изменения
schedule_id - идентификатор расписания
passes - ограничение на количество входов
autostart_days - количество дней автостарта абонеента
start_kind - тип старта абонеента (1- с момента продажи 2 - с момента первого входа)
article - артикул
tax_id - идентификатор налоговой группы

Получение списка финансовых транзакций

путь **/transactions**

метод **GET**

параметры может использоваться любая комбинация параметров:
lo#created_at - начальная дата (YYYY-MM-DDTHH24:MI:SS)
hi#created_at - конечная дата (YYYY-MM-DDTHH24:MI:SS)
paykind_id - вид оплаты
finoperation_id - тип операции
point_name - название торговой точки

ответ

json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Возвращаемые данные:

id - идентификатор транзакции

client_id - идентификатор клиента

account_src_id - идентификатор учетного счета - источника оплаты

account_dest_id - идентификатор учетного счета - получателя оплаты

paykind_id - вид оплаты (0-абонемент 1-депозит 2-наличные 3-кредитная карта 4-банк 5-дополнительная 20-платежные сервисы)

finoperation_id - операция (1-пополнение депозита 2-Оплата абонемента 3-Оплата членского взноса 4-Оплата счета 5-Инкассация 6-Перемещение между счетами)

user_id - id пользователя-автора транзакции

summ - сумма

signed - true-проведена

notes - примечание

point_name - торговая точка

client_surname - фамилия клиента

client_name - имя клиента

paykind_name - название вида оплаты

operation_name - название операции

author_name - имя автора транзакции

Получение списка счетов

путь	/orders
метод	GET
параметры	<p>может использоваться любая комбинация параметров: lo#created_at - начальная дата создания счета (YYYY-MM-DDTHH24:MI:SS) hi#created_at - конечная дата создания счета (YYYY-MM-DDTHH24:MI:SS) lo#payed_at - начальная дата оплаты счета (YYYY-MM-DDTHH24:MI:SS) hi#payed_at - конечная дата оплаты счета (YYYY-MM-DDTHH24:MI:SS) payment_status - признак оплаченности счета (1-не оплачен, 2-оплачен) paykind_id - вид оплаты point_name - название торговой точки</p>
ответ	<p>json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей</p> <p>Возвращаемые данные:</p> <ul style="list-style-type: none"> id - идентификатор счета user_id - id автора счета client_id - идентификатор клиента account_id - учетный счет paykind_id - вид оплаты (0-абонемент 1-депозит 2-наличные 3-кредитная карта 4-банк 5-дополнительная 20-платежные сервисы) point_name - торговая точка created_at - момент создания updated_at - момент изменения table_id - стол guests - количество гостей paykind_name - название вида оплаты author_name - имя автора счета

Получение состава счета

путь	/order_items/id/идентифиатор счета
метод	GET
параметры	идентифиатор счета

ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей
	Возвращаемые данные:
	id - идентификатор позиции счета
	order_id - идентификатор счета
	store_id - идентификатор склада
	good_id - идентификатор товара
	quantity - количество
	good_name - название товара
	summ - сумма позиции счета
	article - артикул
	tax_percent - %НДС
	discount - % скидки
	discount_comment - причина скидки
	task_printed - признак печати встречи
	bringout - номер выноса

Получение списка проходов через СКД

путь	/access_logs
метод	GET
параметры	может использоваться любая комбинация параметров: lo#created_at - начальная дата (YYYY-MM-DDTHH24:MI:SS) hi#created_at - конечная дата (YYYY-MM-DDTHH24:MI:SS)
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка лидов

путь	/leads
метод	GET
параметры	может использоваться любая комбинация параметров: phone#phone-телефон клиента email - электронная почта клиента surname - точная фамилия name - точное имя match#surname - поиск по части фамилии match#name - поиск по части имени
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Создание лида

путь	/leads
метод	POST
параметры	surname - фамилия name - имя info_source_id - источник, из которого добавляется лид phone -номер телефона email - e-mail notes - текст примечания
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка клиентов

путь	/clients
метод	GET
параметры	может использоваться любая комбинация параметров: tag-идентификатор карты клиента phone#phone1-телефон клиента email1 - электронная почта клиента surname - точная фамилия name - точное имя match#surname - поиск по части фамилии match#name - поиск по части имени
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение данных клиента по идентификатору

путь	/clients/id/CLIENT_ID , где CLIENT_ID - идентификатор клиента
метод	GET
параметры	нет
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Создание клиента

путь **/clients**

метод	POST
параметры	<p> surname - фамилия name - имя patronymic - отчество sex - пол (0-жен 1-муж) birthday - дата рождения idcode - идентификационный код city_id - id города postcode - почтовый индекс address - почтовый адрес phone1 - основной номер телефона phone2 - дополнительный номер телефона email1 - основной e-mail email2 - дополнительный e-mail tag - идентификатор карты cardnum - номер карты клиента info_source_id - идентификатор источника информации клиента locked - если не пустая, то это причина блокировки карты клиента message - сообщение клиенту на рецепцию sales_manager_id - id менеджера продаж service_manager_id - id менеджера сервиса allow_delivery - 0-запрещена рассылка клиенту, 1-разрешена iscorporation - 0-частное лицо 1-корпорация corporation_id - id корпорации для частного лица - сотрудника корпорации number_pass - номер паспорта organset_pass - название органа, выдавшего паспорт valid_pass - срок действия паспорта iban - номер счета клиента в банке swift_bic - код банка клиента </p>
ответ	<p> json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей </p>

Изменение данных клиента

путь **/clients/CLIENT_ID, где CLIENT_ID - идентификатор клиента**

метод **PUT**

параметры

- surname** - фамилия
- name** - имя
- patronymic - отчество
- sex** - пол (0-жен 1-муж)
- birthday - дата рождения
- idcode - идентификационный код
- city_id - id города
- postcode - почтовый индекс
- address - почтовый адрес
- phone1 - основной номер телефона
- phone2 - дополнительный номер телефона
- email1 - основной e-mail
- email2 - дополнительный e-mail
- tag - идентификатор карты
- cardnum - номер карты клиента
- info_source_id - идентификатор источника информации клиента
- locked - если не пустая, то это причина блокировки карты клиента
- message - сообщение клиенту на рецепцию
- sales_manager_id - id менеджера продаж
- service_manager_id - id менеджера сервиса
- allow_delivery - 0-запрещена рассылка клиенту, 1-разрешена
- incorporation - 0-частное лицо 1-корпорация
- corporation_id - id корпорации для частного лица - сотрудника корпорации
- number_pass - номер паспорта
- organaset_pass - название органа, выдавшего паспорт
- valid_pass - срок действия паспорта
- iban - номер счета клиента в банке
- swift_bic - код банка клиента

ответ json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Удаление клиента

путь **/clients/CLIENT_ID, где CLIENT_ID - идентификатор клиента**

метод **DELETE**

параметры нет

ответ json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка сотрудников

путь	/staff
метод	GET
параметры	может использоваться любая комбинация параметров: tag-идентификатор карты сотрудника surname - точная фамилия name - точное имя match#surname - поиск по части фамилии match#name - поиск по части имени
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение данных сотрудника по идентификатору

путь	/staff/id/STAFF_ID , где STAFF_ID - идентификатор сотрудника
метод	GET
параметры	нет
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получить абонементы клиента

путь	/client_tickets
метод	GET
параметры	client_id - идентификатор клиента ticket_state - состояние абонемента (1-действующие 2-завершенные 3- не стартованные 4-действующие и нестартованные)
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Оформить абонемент клиенту

путь	/client_tickets
метод	POST
параметры	client_id - идентификатор клиента ticket_id - идентификатор абонеента started - дата начала действия discount - значение скидки discount_kind - тип скидки (0-в процентах 1-в сумме) discount_comment - причина скидки (обязательна, если discount > 0) docnumber - номер договора
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Удалить неоплаченный абонемент клиента

путь	/client_tickets/CLIENT_TICKET_ID , где CLIENT_TICKET_ID - идентификатор абонеента клиента
метод	DELETE
параметры	нет
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Оплата по абонементу

путь	/transactions
метод	POST
параметры	account_dest_id - идентификатор счета абонеента клиента (account_summ_id) client_id - идентифкатор клиента finoperation_id - идентификатор операции. Оплата по абонементу = 2 paykind_id - вид расчетов (1-депозит 2-наличные 3-кредитная карта 4-безналичные 5-дополнительная форма) summ - сумма оплаты
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Оплата членского взноса

путь	/transactions
метод	POST
параметры	account_dest_id - идентификатор счета членского взноса абонемента клиента (account_membership_id) client_id - идентификатор клиента finoperation_id - идентификатор операции. Оплата членского взноса = 3 paykind_id - вид расчетов (1-депозит 2-наличные 3-кредитная карта 4-безналичные 5-дополнительная форма) summ - сумма оплаты point_name - название точки создания транзакции
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение депозитных счетов клиента

путь	/client_accounts
метод	GET
параметры	client_id - идентификатор клиента
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Пополнение депозитного счета

путь	/transactions
метод	POST
параметры	account_dest_id - идентификатор депозитного счета client_id - идентификатор клиента finoperation_id - идентификатор операции. Пополнение депозита = 1 paykind_id - вид расчетов (1-депозит 2-наличные 3-кредитная карта 4-безналичные 5-дополнительная форма) summ - сумма оплаты point_name - название точки создания транзакции
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка групповых занятий

путь	/lesson_calendars/feed
метод	GET
параметры	start - дата-время начала диапазона end - дата-время конца диапазона
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Запись клиента на групповое занятие

путь	/lesson_calendar_bookings
метод	POST
параметры	client_id - идентификатор клиента lesson_calendar_id - идентификатор занятия
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка записанных на занятие

путь	/lesson_calendar_bookings
метод	GET
параметры	lesson_calendar_id - идентификатор занятия
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей

Получение списка филиалов

путь	/branches
метод	GET
параметры	нет
ответ	<p>json-объект со свойствами: success - признак успешности, error - текст ошибки, data - массив данных, total - количество записей</p> <p>id - идентификатор филиала</p> <p>Атрибуты данных:</p> <p>name - название</p> <p>admin_name - ответственное лицо</p> <p>admin_email - email ответственного</p> <p>admin_phone - телефон ответственного</p> <p>country - название страны</p> <p>country_geoid - код страны по базе geoid</p> <p>region - название региона</p> <p>region_geoid - код региона по базе geoid</p> <p>city - город</p> <p>city_geoid - код города по базе geoid</p> <p>latitude, longitude - координаты</p> <p>lang_id - id языка интерфейса по-умолчанию</p> <p>address - адрес филиала</p> <p>phones - телефоны филиала</p> <p>email - email филиала</p> <p>website - сайт филиала</p> <p>timezone - временная зона</p> <p>uuid - альтернативный идентификатор филиала</p> <p>currency - код валюты</p> <p>country_code - код страны</p> <p>resource_id - id ресурса логотипа филиала</p>

Функции авторизации клиентов

Функции предназначены для проверки авторизации клиентов. Авторизация основана на номере телефона и личном пароле. Функции авторизации клиентов могут быть полезны для интеграции с сайтом клуба или системой самообслуживания.

Преавторизация клиента

Преавторизация предназначена для проверки наличия клиента с таким номером телефона. Если преавторизация выполняется впервые для клиента, то для проверки номера клиента, функция генерирует код авторизации, который отправляется на электронную почту клиента. Если в анкете клиента нет электронной почты, то код авторизации отправляется в смс.

путь	/clients/preauthorize
метод	POST
параметры	phone - номер телефона клиента
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - объект данных, total - количество записей data содержит: client_id - идентификатор клиента first_login - true - признак того, что пароль для клиента еще не задан, необходимо провести проверку кода активации surname - фамилия клиента name - имя клиента message - сообщение клиенту - один из вариантов («Код активации отправлен в СМС на номер» или «Введите пароль»)

Проверка кода авторизации

путь	/clients/:id/check_activation_code , где :id - идентификатор клиента
метод	POST
параметры	code - код активации для проверки
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - объект данных, total - количество записей

Установка пароля для клиента

путь	/clients/:id/set_password , где :id - идентификатор клиента
метод	POST
параметры	password - пароль
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - объект данных, total - количество записей

Проверка пароля

путь	/clients/:id/check_password , где :id - идентификатор клиента
метод	POST
параметры	password - пароль
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - объект данных, total - количество записей

Сброс пароля

путь	/clients/:id/reset_password, где :id - идентификатор клиента
метод	POST
параметры	нет
ответ	json-объект со свойствами: success - признак успешности, error - текст ошибки, data - объект данных, total - количество записей

Аутентификация клиента через OAuth

Поддерживается функция аутентификации клиентов, которая выполняется на стороне LuckyFit с перенаправлением на ваш ресурс после успешного выполнения.

путь	/oauth
метод	GET
параметры	redirect - url страницы для перехода
ответ	После успешной авторизации перенаправление на redirect передачей параметра client_id - идентификатор клиента в LuckyFit. В случае отказа клиента в аутентификации происходит переход на redirect без передачи параметра client_id

Оформление покупки абонеента с оплатой кредитной картой

путь	/payment_systems/pay/order_ticket?client_id=&ticket_id=&summ=
метод	GET
параметры	client_id - идентификатор клиента, ticket_id - идентификатор продаваемого абонеента из ассортимента, summ - стоимость абонеента (если не задана, то стоимость по прайсу)
ответ	страница оплаты

Данный метод не требует авторизации, он отображает платежную страницу для заполнения данных платежа и после успешной оплаты клиенту оформляется абонеент.

Оплата по счету с оплатой кредитной картой

путь	/payment_systems/pay/account?id=&summ=
------	--

метод	GET
параметры	id - идентификатор счета, который необходимо пополнить. summ - сумма оплаты
ответ	страница оплаты