API HTTP-сервиса «Фестивали»

Общая информация

- HTTP-сервис: **Фестивали**
- RootURL: `fest`
- Основной маршрут обмена: PUT `/v1/{ИмяМетода}`
- Технически в публикации 1С обычно используется путь вида `/hs/fest/v1/{ИмяМетода}`.

Общие правила для PUT /v1/{ИмяМетода}

- Тело запроса: JSON.
- Обязательное поле тела: ВерсияПриложения.
- Если версия ниже допустимой, возвращается JSON:
  - `Удачно = false`
  - Ошибка = «Необходимо обновить приложение...»
  - СсылкаНаНовуюВерсию = «https://rusclimbing.ru/climbfest/»
- Успешный HTTP-код обычно `200`, бизнес-ошибки идут в JSON-поле Ошибка.

Команды

Авторизация и профиль

check_email

- Назначение: проверка существования пользователя по email.
- Вход:
  - ЭлектроннаяПочта
- Выход JSON:

```
{
  "Удачно": true,
  "Ошибка": "",
  "ПользовательСуществует": true
}
```

new_email

- Назначение: создание пользователя по email.
- Вход:
  - ЭлектроннаяПочта
- Выход JSON:

```
{
  "Удачно": true,
  "Ошибка": ""
}
```

auth_email

- Назначение: подтверждение кода и выдача токена.
- Вход:
  - ЭлектроннаяПочта
  - КодПроверки
- Выход JSON:

```
{
  "Удачно": true,
  "Ошибка": "",
  "ПользовательАвторизован": true,
  "Токен": "..."
}
```

- Нюанс реализации: в одной из веток может вернуться JSON `false` вместо структуры.

Списки/справочники

get_festlist

- Вход: без дополнительных полей.
- Выход JSON: {Удачно, Ошибка, Фестивали:[...]}.
- Поля элемента Фестивали:
  - Код, Наименование, Начало, Окончание, Место, ДатыДляКалендаря, ОписаниеДляПриложения, СетыПослеОплаты, Страница
  - Оферта, СогласиеПДн, СетОбязателен, НесколькоЗачетов, РасчетРезультатовНаСервере
  - РазмерГруппы, КлубныйЗачет, СемейныйЗачет, СтуденческийЗачет
  - ИспользоватьУровеньЛазания, УровеньЛазанияОбязателен, ИспользоватьСтаж, СтажОбязателен
  - ИспользоватьКласс, КлассОбязателен, ИспользоватьГород, ГородОбязателен
  - ИспользоватьТелефон, ТелефонОбязателен, ИспользоватьРазряд, ИспользоватьБылНаМесте
  - ИспользоватьПакеты, ИспользоватьПромокоды, УчитыватьПопытки, УчитыватьФлешРедпойнт

get_linklist

- Вход: Фестиваль (код).
- Выход: {Удачно, Ошибка, Ссылки:[{Код, Наименование, АдресСсылки, Порядок}]}.

get_regionlist

- Вход: без дополнительных полей.
- Выход: {Удачно, Ошибка, Регионы:[{Код, Наименование}]}.

get_categorylist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Зачеты:[…]}.
- Поля Зачеты: Код, Наименование, Порядок, НуженСопровождающий, КоличествоСетов, ЦенаПоСетам, ТрассТрудность, ТрассБоулдеринг, ТрассМультипитч, ЗолотоМ, СереброМ, БронзаМ, ЗолотоЖ, СереброЖ, БронзаЖ.

get_packagelist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Пакеты:[{Код, Наименование, Порядок, НуженРазмер}]}.

get_discountlist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Скидки:[{Код, Наименование, Скидка, Порядок, ПоПромокоду}]}.

get_doplist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Допы:[{Код, Наименование, Цена, ЕстьСвойство, НаименованиеСвойства}]}.

get_questlist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Вопросы:[…]}.
- Поля Вопросы: Код, Наименование, Порядок, Описание, Обязательный, ДопускаетсяНесколькоОтветов, ВариантыОтветов.

get_optionlist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, СвойстваДопов:[{Код, Наименование, ДопКод}]}.

get_doctypelist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, ВидыДокументов:[{Код, Наименование, Порядок}]}.

get_levellist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, УровниЛазания:[{Код, Наименование}]}.

get_experiencelist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Стажи:[{Код, Наименование}]}.

get_classlist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Классы:[{Код, Наименование}]}.

get_setlist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, Сеты:[...]}.
- Поля Сеты: Код, Наименование, Начало, СвободныхМест, Зачеты, Цена.

get_sizelist

- Вход: без дополнительных полей.
- Выход: {Удачно, Ошибка, Размеры:[{Код, Наименование}]}.

Проверки и расчёты

check_category

- Вход:
  - Фамилия, Имя, Отчество, ДатаРождения
  - Фестиваль, УровеньЛазания, Стаж
- Выход:

```
{
  "Удачно": true,
  "Ошибка": "",
  "Зачет": "...",
  "Скидка": "...",
  "Сет": "...",
  "Пакет": "..."
```

```
}
```

check_promo

- Вход: Промокод, Фестиваль.
- Выход: {Удачно, Ошибка, Принят, Скидка}.

get_amount

- Вход: ТокенПользователя, Код.
- Выход: {Удачно, Ошибка, Сумма}.

get_groupamount

- Вход: ТокенПользователя, Фестиваль.
- Выход: {Удачно, Ошибка, Сумма}.

get_preamount

- Вход: ТокенПользователя, Фестиваль, Пакет, Зачет.
- Опционально: Скидка, Сеты (массив кодов).
- Выход: {Удачно, Ошибка, Сумма}.

Участники

add_participant

- Назначение: создание участника.
- Основные входные поля:
  - ТокенПользователя, Фестиваль, Регион
  - Фамилия, Имя, Отчество, НетОтчества, ДатаРождения, Пол
  - Сопровождающий1, Сопровождающий2, ЭтоСтудент
  - ЭтоЛидерСемьи, СемьяНазвание, ЭтоЛидерКлуба, КлубНазвание
  - Сет1, Сет2
- Дополнительные/условные поля:
  - Код, Пакет, УровеньЛазания, Стаж, Класс, Разряд, РазмерОдежды
  - Сет3, Сет4, Сет5, Промокод, Телефон, Город, БылНаМесте
  - СемьяКод, ПарольСемьи, КлубКод, ПарольКлуба, Ответы
- Выход: {Удачно, Ошибка, Код}.

edit_participant

- Назначение: изменение участника (тот же обработчик, что `add_participant`).
- Важно: для редактирования обязателен Код участника.

- Выход: {Удачно, Ошибка, Код}.

add_participant_bycode

- Вход: ТокенПользователя, Фестиваль, Код, Пароль.
- Выход: {Удачно, Ошибка, Код}.

delete_participant

- Вход: ТокенПользователя, Код.
- Выход: {Удачно, Ошибка, Код}.

get_participantlist

- Вход: ТокенПользователя, Фестиваль.
- Выход: {Удачно, Ошибка, Участники:[...]}.
- Поля элемента Участники:
  - Код, Фамилия, Имя, Отчество, Пол, ДатаРождения, ГодРождения, Регион, Город, Телефон
  - Разряд, БылНаМесте, КатегорияТрудность, КатегорияБоулдеринг
  - Зачет, Пакет, Скидка, СтатусСкидки, ЭтоСтудент, СлотРегистрации
  - Оплачен, Проверен, ОшибкаПроверки, Допущен
  - Сопровождающий1, Сопровождающий2, Сет1, Сет2, Сет3, Сет4, Сет5, Пароль
  - УровеньЛазания, Стаж, Класс
  - ЭтоЛидерСемьи, СемьяНазвание, СемьяПароль, СемьяКод
  - ЭтоЛидерКлуба, КлубНазвание, КлубПароль, КлубКод

get_paylink

- Вход: ТокенПользователя, Код.
- Выход: {Удачно, Ошибка, Ссылка}.

get_grouppaylink

- Вход: ТокенПользователя, Фестиваль.
- Выход: {Удачно, Ошибка, Ссылка}.

Трассы и результаты

get_routelist

- Вход: ТокенПользователя, КодУчастника, Пароль.
- Выход: {Удачно, Ошибка, Трассы, КатегорииТрасс}.
- Поля элемента Трассы:
  - Код, Наименование, Номер, ТипКод, КатегорияКод, ЕстьЗона

    - Баллы, БаллыРедпойнт, БаллыЗона
    - КодГруппы, ЭтоГруппа
    - ДатаПролаза, ПролезТолькоЗону, МультипитчВерхняяСтраховка, МультипитчИТО, ПопытокТоп, ПопытокЗона

ЕстьЗона:

- Истина - на трассе есть зона, необходимо дать возможность отметить пролезание зоны (если не вылез всю трассу);
- Ложь - на трассе нет зоны.

ТипКод:

- 000000001 - Боулдеринг (Б)
- 000000002 - Трудность (Т)
- 000000003 - Мультипитч (М)

Если трасса мультипитча, то необходимо показывать переключатель (флажок) **Использовал ИТО или завис** (сохранять в МультипитчИТО)

Реквизит МультипитчВерхняяСтраховка на текущий момент не используется.

Реквизиты Баллы, БаллыРедпойнт, БаллыЗона при расчете баллов на сервере не используются.

ЭтоГруппа:

- Истина - данный элемент списка является группой (не трассой), содержащей другие группы или трассы;
- Ложь - данный элемент списка является трассой.

КодГруппы: код элемента-группы, в которую входит текущий элемент.

Если заполнено ДатаПролаза - есть какой-то результат:

- ПролезТолькоЗону = Истина - пролез только зону (не всю трассу);
- ПролезТолькоЗону = Ложь - TOP (пройдена вся трасса).

Реквизиты ПопытокТоп и ПопытокЗона необходимо отображать и вводить только в случае, если в реквизитах фестиваля **УчитыватьПопытки = Истина**. В противном случае можно отправлять пустые значения.

save_results

- Вход:
    - ТокенПользователя, КодУчастника, Пароль
    - Трассы (массив)
- Поля элемента Трассы:
    - КодТрассы, ДатаПролаза, МультипитчВерхняяСтраховка, МультипитчИТО, ПролезТолькоЗону, ПопытокТоп, ПопытокЗона
- Выход: {Удачно, Ошибка}.

save_oneresult

- Вход:
  - ТокенПользователя, КодУчастника, Пароль
  - Трасса (объект)
- Поля Трасса:
  - КодТрассы, Пролез, ДатаПролаза, МультипитчВерхняяСтраховка, МультипитчИТО, ПролезТолькоЗону, ПопытокТоп, ПопытокЗона
- Выход: {Удачно, Ошибка}.

Аренда снаряжения

get_equiplist

- Вход: Фестиваль.
- Выход: {Удачно, Ошибка, СписокСнаряжения}.
- Поля элемента СписокСнаряжения:
  - Код, Наименование, Описание
  - ПериодыСтрокой - строка - перечень свободных периодов (дней), на которые можно заказать это снаряжение, разделенных точкой с запятой (например, «10.06;11.06;15.06»)
  - Цены - список цен (упорядочен по возрастанию МинимумПериодов):
    - МинимумПериодов - при заказе на сколько дней действует цена;
    - Цена - значение цены.

get_equiporderlist

- Вход: ТокенПользователя, Фестиваль.
- Выход: {Удачно, Ошибка, Заказы}.
- Поля элемента Заказы:
  - Номер, Дата, Фестиваль, Сумма, Пароль
  - Состояние (пустая строка - не оплачен, 000000001 - оплачен, 000000002 - выдан, 000000003 - возвращен)
  - Состав заказа - список снаряжения и периодов в заказе:
    - НомерСтроки, Снаряжение, Цена, Сумма
    - Периоды - выбранные периоды аренды, разделенные точкой с запятой (например, «10.06;11.06;15.06»)
    - Количество - количество выбранных периодов аренды

From:
http://127.0.0.1:8000/doc/ - **Документация портала ФСР**

Permanent link:
http://127.0.0.1:8000/doc/climbfest/api?rev=1778415109

Last update: **10.05.2026 12:11**