

# 1. Общая информация

---

API состоит из методов на одной стороне сторонах: FuelUp.

Базовый адрес на стороне FuelUp (для передачи статусов и прочего) отличается для тестовой системы и для рабочей:

1. Тестовый сервер, базовый адрес `https://terminal.api.dev.fuelup.ru/`
2. Рабочий сервер, базовый адрес `https://terminal.api.fuelup.ru/`

Чтобы вызвать любой из методов на стороне FuelUp, необходимо выполнить запрос по адресу `<базовый адрес>/<метод>`.

Если запрос не проходит валидацию, возвращается HTTP-код `400 Invalid request`.

Если всё прошло успешно, сервер отвечает кодом `200 OK`.

Если возвращаются любые другие HTTP-ошибки, значит запрос не был обработан сервером и нужно повторить его позднее.

## 2. Данные, необходимые для обмена

---

1. `api_key` - секретный ключ, используется идентификации партнера, генерируется FuelUp.

Перечисленные параметры доступны для просмотра/изменения в личном кабинете FuelUp.

## 3. Методы, выполняемые на стороне FuelUp

---

### 3.1. Передача конфигурации АЗС

---

Для передачи текущей конфигурации АЗС (кол-во и статус ТРК, типы топлива доступные на ТРК) на сервер FuelUp, АСУ АЗС периодически и в момент обновления конфигурации должна присылать HTTP запрос:

URL: `<базовый URL>/api/station`

HEADER: Authorization: {api\_key}

POST-данные в формате JSON:

```

{
  "type": "object",
  "additionalProperties": false,
  "required": ["StationExtendedId", "Columns"],
  "properties": {
    "StationExtendedId": {
      "description": "Уникальный внешний идентификатор станции АЗС",
      "type": "string"
    },
    "Columns": {
      "description": "Список ТРК на АЗС. Ключ объекта - номер ТРК на АЗС",
      "type": "object",
      "additionalProperties": false,
      "items": {
        "1": {
          "description": "Описание ТРК <1>",
          "type": "object",
          "required": ["Fuels"],
          "properties": {
            "Fuels": {
              "description": "Список доступных типов топлива на ТРК <1>",
              "type": "array",
              "additionalProperties": false,
              "items": {
                "description": "Идентификатор типа топлива",
                "type": "string"
              }
            }
          }
        },
        "N": {
          "description": "Описание ТРК <N>",
          "type": "object",
          "required": ["Fuels"],
          "properties": {
            "Fuels": {
              "description": "Список доступных типов топлива на ТРК <N>",
              "type": "array",
              "additionalProperties": false,
              "items": {
                "description": "Идентификатор типа топлива",
                "type": "string"
              }
            }
          }
        }
      }
    }
  },
  "Enable": {

```

```
    "description": "Статус станции: true - доступна, false - выключена",
    "type": "string",
    "enum": ["true", "false"]
  },
  "Name": {
    "description": "Наименование станции",
    "type": "string"
  },
  "Address": {
    "description": "Адрес станции",
    "type": "string"
  },
  "Location": {
    "description": "Гео-координаты станции",
    "type": "object",
    "additionalProperties": false,
    "required": ["Lat", "Lon"],
    "properties": {
      "Lat": {
        "description": "Широта",
        "type": "numeric"
      },
      "Lon": {
        "description": "Долгота",
        "type": "numeric"
      }
    }
  },
  "FuelNames": {
    "description": "Фирменное название видов топлива",
    "type": "array",
    "additionalProperties": false,
    "items": {
      "description": "Идентификатор типа топлива",
      "type": "object",
      "required": ["Fuel", "FuelName"],
      "properties": {
        "Fuel": {
          "description": "Идентификатор типа топлива",
          "type": "string"
        },
        "FuelName": {
          "description": "Фирменное название типа топлива",
          "type": "string"
        }
      }
    }
  }
}
```

Возможные идентификаторы топлива

ID	Марка
diesel	дизель
diesel_premium	брендированный дизель
a80	бензин марки A80
a92	бензин марки A92
a92_premium	брендированный бензин марки A92
a95	бензин марки A95
a95_premium	брендированный бензин марки A95
a98	бензин марки A98
a98_premium	брендированный бензин марки A98
a100	бензин марки A100
a100_premium	брендированный бензин марки A100
propane	газ пропан
metan	метан

Пример

```
{
  "StationExtendedId": "1234",
  "Columns": {
    "1": {
      "Fuels": ["a92", "a95", "a95_premium", "diesel_premium"]
    },
    "2": {
      "Fuels": ["a92", "a95", "a95_premium", "a100_premium"]
    }
  },
  "Enable": true,
  "Name": "АЗС \"На Мандариновой\" №10",
  "Address": "г. Энгс, ул. Мандариновая, 4",
  "Location": { "Lon": 43.5432, "Lat": 34.1234 },
  "FuelNames": [
    {
      "Fuel": "a92",
```

```

        "FuelName": "АИ-92"
    },
    {
        "Fuel": "a95",
        "FuelName": "АИ-95"
    },
    {
        "Fuel": "a95_premium",
        "FuelName": "Г-95"
    },
    {
        "Fuel": "a100_premium",
        "FuelName": "Г-100"
    },
    {
        "Fuel": "diesel_premium",
        "FuelName": "ДТ Опти"
    }
]
}

```

## 3.2. Передача прайс-листа

Для передачи прайс-листа на сервер FuelUp, АСУ АЗС периодически и в момент обновления прайс-листа должна присылать HTTP запрос:

URL: <базовый URL>/api/price

HEADER: Authorization: {api\_key}

POST-данные: key1=value1&key2=value2&...

где:

key – идентификатор топлива

value – цена за 1 литр (тип double, через точку, например, 43.95)

### Пример

a95=44.95&a100=55.90

## 3.3. Обработка заказов

АСУ АЗС должна постоянно опрашивать сервер FuelUp для проверки поступления новых заказов. Для этого необходимо выполнить HTTP запрос:

URL: <базовый URL>/api/orders/items

HEADER: Authorization: {api\_key}

В случае ответа 200 OK при наличии заказов тело ответа будет следующего вида:

## Схема ответа:

```
{
  "type": "object",
  "additionalProperties": false,
  "required": ["Orders", "nextRetryMs"],
  "properties": {
    "nextRetryMs": {
      "description": "Интервал в миллисекундах - через какое время допустимо повтор",
      "type": "integer"
    },
    "Orders": {
      "description": "Список новых заказов",
      "type": "array",
      "items": {
        "description": "Параметры заказа",
        "type": "object",
        "additionalProperties": false,
        "required": ["Id", "DateCreate", "OrderType", "OrderVolume", "ColumnId",
        "properties": {
          "Id": {
            "description": "Уникальный идентификатор заказа в системе FuelUp",
            "type": "string"
          },
          "DateCreate": {
            "description": "Дата и время создания заказа в UTC",
            "type": "datetime"
          },
          "OrderType": {
            "description": "Тип заказа (сумма, объем литров, полный бак)",
            "type": "string",
            "enum": ["Money", "Liters", "FullTank"]
          },
          "OrderVolume": {
            "description": "Значение заказа",
            "type": "numeric"
          },
          "StationId": {
            "description": "Идентификатор АЗС в системе партнёра",
            "type": "string"
          },
          "StationExtendedId": {
            "description": "Внешний идентификатор АЗС",
```



1. `AcceptOrder` – заказ создан, полностью оплачен и ожидает подтверждения от АСУ АЗС
2. `Expire` – статус от АЗС не поступил в течение 30 минут
3. `Completed` – заказа завершен успешно
4. `StationCanceled` – заказ отменен оператором АЗС или же интегрируемой системой
5. `UserCanceled` – заказ отменен пользователем
6. `WaitingRefueling` – ожидание включения налива на ТРК
7. `Fueling` – идет налив

При получении заказа в статусе `OrderCreated` интегрируемая система должна проверить ряд параметров:

1. `ColumnId` – в случае, если идентификатор колонки не найден, либо колонка отключена, АСУ АЗС должна сообщить статус `Canceled` с указанием соответствующей причины отмены.
2. `FuelId` – в случае, если на указанной колонке заданное в заказе топливо недоступно, АСУ АЗС должна сообщить статус `Canceled` с указанием соответствующей причины отмены.
3. `PriceFuel` – в случае, если стоимость топлива в интегрируемой систем отличается от присланной, АСУ АЗС должна сообщить статус `Canceled` с указанием соответствующей причины отмены.

Сервер FuelUp фиксирует, когда был отправлен последний запрос на получение заказов. В случае если прошло больше 5 минут с момента последнего запроса на получение списка заказов, станция временно выключается до момента поступления запроса на получение заказа.

**Заказы в АСУ АЗС необходимо создавать на литры, параметр `Litre`**

### Пример

```
{
  "nextRetryMs": 2000,
  "Orders": [
    {
      "Id": "088ccc98-73e3-4b4a-aaa9-79dddd02720a",
      "DateCreate" : "2019-11-20T17:28:33.994Z",
      "OrderType": "Litre",
      "OrderVolume": "10",
      "StationId": "1234",
      "StationExtendedId": "1234",
      "ColumnId": "1",
      "FuelId": "a92",
      "PriceFuel": "42.50",
      "Litre": "10",
      "Sum": "425.00",
      "Status": "OrderCreated",
      "ContractId": "32135"
    }
  ]
}
```



```
}  
  ]  
}
```

## 4. Передача статусов заказов на FuelUp

---

### 4.1. Статус Аксепт

---

Данный статус сообщает FuelUp о том, что заказ принят и обработан интегрируемой системой.

Данный статус интегрируемая система отправляет после того, как были произведены некоторые действия с заказом (например, заказ сохранен в базе данных) и система готова перейти на следующий шаг.

В случае если FuelUp дала ответ отличный от `200 OK`, то интегрируемая система отправляет статус `Canceled` и прекращает обработку заказа.

GET, URL: `<базовый URL>/api/orders/accept?orderId={orderId}`

HEADER: `Authorization: {api_key}`

Все параметры являются обязательными.

После отправки статуса Аксепт заказ перейдет в статус `waitingRefueling`

### 4.2. Статус Fueling

---

Данный статус сообщает FuelUp о том, что интегрируемая система готова запустить колонку (начать пролив). Статус отправляется перед началом пуска колонки.

В случае если FuelUp дала ответ отличный от `200 OK`, интегрируемая система отправляет статус `Canceled` и прекращает обработку заказа.

GET, URL: `<базовый URL>/api/orders/fueling?orderId={orderId}`

HEADER: `Authorization: {api_key}`

Все параметры являются обязательными.

### 4.3. Статус Canceled

---

Данный статус сообщает FuelUp о том, что заказ следует отменить.

GET, URL: `<базовый URL>/api/orders/canceled?orderId={orderId}&reason={reason}&extendedOrderId={extendedOrderId}&extendedDate={extendedDate}`

HEADER: Authorization: {api\_key}

`reason` – текстовая причина отмены

`extendedOrderId` – идентификатор заказа в АСУ сети

`extendedDate` – дата, по которой АСУ АЗС строит отчет для сверки, формат `dd.MM.yyyy HH:mm:ss`

Все параметры являются обязательными.

## 4.4. Статус Completed

---

Данный статус сообщает FuelUp о том, что заказ выполнен и топливо залито.

В случае если FuelUp дала ответ отличный от `200 OK`, интегрируемая система отправляет запросы с некоторым отложенным кол-вом времени до момента получения ответа `200 OK`.

GET, URL: `<базовый URL>/api/orders/completed?orderId={orderId}&litre={litre}&extendedOrderId={extendedOrderId}&extendedDate={extendedDate}`

HEADER: Authorization: {api\_key}

POST-данные: `orderId={orderId}&litre={litre}&extendedOrderId={extendedOrderId}&extendedDate={extendedDate}`

`litre` – кол-во пролитых литров, указывается как `double` с разделителем точка

`extendedOrderId` – идентификатор заказа в АСУ сети

`extendedDate` – дата, по которой АСУ строит отчет для сверки, формат `dd.MM.yyyy HH:mm:ss`

Все параметры являются обязательными.

## 4.5. Отправка счетчика налива

---

В момент процесса налива интегрируемая система может сообщать FuelUp статус счетчика налива.

Для этого необходимо с периодичностью от 5 – 10 секунд выполнять запрос (для исключения повышенной нагрузки в АСУ АЗС следует устанавливать timeout на запрос не более 1 секунды (1000 мс))

URL: `<базовый URL>/api/orders/volume`

HEADER: Authorization: {api\_key}

POST-данные: `orderId={ordeId}&litre={litre}`

`litre` – кол-во пролитых на момент отправки запроса литров, указывается как double с разделителем точка

Все параметры являются обязательными.

## 4.6. Отправка сообщений пользователю

---

В момент обработки заказа интегрируемая система может направлять произвольные нотификации клиенту в приложение.

Для отправки сообщений необходимо выполнить запрос

URL: `<базовый URL>/api/orders/message`

HEADER: Authorization: {api\_key}

POST-данные: `orderId={ordeId}&msg={msg}`

`msg` – текст сообщения, которое будет показано в приложение

Все параметры являются обязательными.

