

## Описание API приложения AZS-Online v2.0 версия документа 1.0

### Общее описание протокола:

Сервис AZS-ONLINE позволяет выгружать данные, производить управление получателями, лимитами и резервуарами подключенных ТРК (в различные программы например 1С) посредством API.

Взаимодействие с сервисом AZS-ONLINE (<https://garvekswebreporter.ru>) происходит посредством REST протокола, промежуточные состояния между запросами не сохраняются и должны сохраняться если требуется на стороне клиента (программы) использующей API.

Для защиты данных используется доступ по протоколу HTTPS.

API использует HTTP методы GET, POST, PUT, DELETE, PATCH для различия выполняемых действий.

Формат данных протокола JSON.

Для доступа к API необходимо иметь действующий логин и пароль от личного кабинета аккаунта.

В описании методов указаны адреса конечных точек запроса и их формат.

Для тестирования функций API и получения примеров написания кода для работы с запросами рекомендуем использовать программу POSTMAN (<https://www.postman.com>).

## Доступные возможности в зависимости от версии API:

**Версия API 1.0 - 1.2 позволяет доступ только для аккаунтов Администраторов:**

1. Получение списка ТРК привязанных к аккаунту
2. Получение событий со всех ТРК привязанных к аккаунту
3. Создание, редактирование, удаление Получателей, запись и удаление их с ТРК
4. Создание, редактирование, удаление фирм(групп получателей)
5. Получение списка, корректировка и пополнение резервуаров привязанных к аккаунту
6. Доступ к журналам пополнения резервуаров
7. Создание, редактирование, удаление поставщиков ГСМ

**Версия API 2.0 – разрешает ограниченный доступ к данным пользователям:**

1. Получение событий ограниченных правами доступа аккаунта к ТРК и фирмам соответственно.

## Указание версии API в запросе:

Для указания версии запрашиваемого API необходимо к адресу запроса добавить параметр:

?api-version=2 — для версии API 2.0 (в данный момент работает только для получения событий )

**для API версии 1.0 — 1.1 указывать параметр версии не нужно!!!**

### Получение токена:

Для получения токена нужно выполнить POST запрос на адрес -

<https://garvekswebreporter.ru/token>

С заголовком запроса:

Content-Type:application/x-www-form-urlencoded

Включив тело запроса:

grant\_type:password

username:Ваш логин

password:Ваш пароль

**Логин и пароль должны быть в формате UriEncode!!!**

В ответ сервер вернет вам токен доступа в формате JSON (пример):

```
{
  "access_token":
  "dBBQK8rtJU0SGWTOwKdw6hH0EchuAKSMTgv7efgZwhbRypKUopKCQdgHjbGQqRgThszRoyQSxLmA ",
  "token_type": "bearer",
  "expires_in": 86399,
  "userName": "c***s@yaex.ru",
  ".issued": "Wed, 18 Oct 2017 12:51:02 GMT",
  ".expires": "Thu, 19 Oct 2017 12:51:02 GMT"
}
```

Все дальнейшие запросы нужно выполнять, включая этот токен в заголовок запросов (пример):

Authorization: Bearer SSRDgb9WTpFLyO0HGfTCdl2IyY7Uouh8qe9f8P6-cWA1S\_Fqsy6d0IG5-h

Срок действия токена 24 часа с момента получения, после его истечения нужно запросить новый токен.

Accept: application/json            - JSON формат

## Получение событий реализация V1.0:

### Запрос событий за текущий день:

GET <https://garvekswebreporter.ru/api/events>

Запрос событий за указанную дату:

GET <https://garvekswebreporter.ru/api/events/?time=2017-10-11>

Формат времени time : yyyy-mm-dd.

### Структура ответа на запрос:

```
{
  "NumPetrol": 9,
  "NameKolodka": "865209030715065",
  "DataTime": "2017-10-11T15:10:00",
  "NameUser": "Диспетчер Колонки",
  "IdUser": "0",
  "Firm": "Оператор",
  "NumTc": "",
  "Odometr": 0,
  "Waybill": 0,
  "AmountFuel": 4.17,
  "Idkolodka": "865209030715065"
}
```

- порядковый номер заправки
- имя колонки
- время и дата заправки
- имя и фамилия водителя
- номер карты водителя
- наименование фирмы
- номер тс
- одометр
- путевой лист
- количество выданного топлива в литрах
- уникальный идентификатор колонки

## Получение событий по последнему номеру заправки на указанной колонке:

POST <https://garvekswebreporter.ru/api/events>

```
{
  "IdKolonka": "864287035715745",      - ID колонки
  "NumPetrol": 112                    - Номер запрашиваемой заправки
}
```

Вернет заправки начиная с запрошенного (включительно) номера:

## Структура ответа на запрос:

```
{
  "Id": 198094,                        - Уникальный ID события
  "IdKolonka": "864287035715745",     - Id колонки
  "NumPetrol": 112,                   - номер заправки
  "IdUser": "7439",                  - Id карты пользователя
  "NameUser": "ПРОВЕРКА ЛИМИТА",      - Имя и фамилия водителя (разделитель пробел)
  "Firm": "ТЕСТ",                   - Название фирмы
  "IdTc": null,*                      - ID двойной идентификации (по умолчанию IdUser)
  "NumTc": "",                       - Номер ТС (по умолчанию пустая строка)
  "Odometr": 0,                      - Одометр (по умолчанию 0)
  "Waybill": 0,                      - Путевой лист (по умолчанию 0)
  "DateTime": "2018-03-30T16:05:00",  - Время начала заправки
  "AmountFuel": 0.06,                - Выдано
  "NameKolonka": "Демо",              - Имя колонки
  "Login": "7806523282",              - Id личного кабинета администратора
  "UserOmnicomId": 236016989,        - Id Omnicomт если не привязан 0
  "UnixTimeStamp": 1522415100,        - Unix-время начала заправки
  "Comment": null,*                  - Комментарий к событию (всегда null)
  "Coast": 0,*                        - Цена заправки (всегда 0 )
  "UnixTimeStampEnd": 1522415100,*    - Unix-время окончания заправки
  "DateTimeEnd": "2018-03-30T16:05:00"* - Время окончания заправки
}
```

\*- Поля с текущими версиями прошивки КВТ не используются.

UnixTimeStampEnd, DateTimeEnd - принимают значения времени загрузки события на сервер. В будущих версиях прошивок они будут принимать значения окончания заправки.

## Получение событий реализация V2.0:

Запрос событий за текущий день без фильтрации результатов:

GET <https://garvekswebreporter.ru/api/events?api-version=2>

Структура ответа на запрос:

```
{
  "Id": 843342,                - уникальный номер записи
  "IdKolonka": "867793030485256",
  "NumPetrol": 1202,
  "IdUser": "0",
  "NameUser": "Заправка отменена получателем",
  "Firm": "Система",
  "IdTc": "0",
  "NumTc": "-",
  "Odometr": 0,
  "Waybill": 0,
  "DateTime": "2020-02-18T11:22:46",
  "AmountFuel": 0.0,
  "NameKolonka": "867793030485256",
  "Login": "7806529882",
  "UserOmnicomId": 0,
  "UnixTimeStamp": 1582014166,
  "UnixTimeStampEnd": 1582014166,
  "DateTimeEnd": "2020-02-18T11:22:46",
  "NameTank": "ТЕСТОВЫЙ",    - название резервуара или секции
  "NumPost": 1,               - номер поста
  "StartTankLevel": 0.0,      - уровень в резервуаре на начало события
  "EndTankLevel": 0.0,        - уровень в резервуаре на конец события
  "Density": 0.0,             - плотность
  "AmountMass": 0.0,          - масса (пока не используется)
  "Coast": 0.00,
  "Section": 0,               - номер секции (резервуара)
  "PetrolOption": 0,          - опции заправки (пока не используется)
  "Longitude": 0.0,           - Долгота (координаты)
  "Latitude": 0.0,            - Широта (координаты)
  "Comment": null
}
```

## Получение событий с фильтрацией результатов:

POST <https://garvekswebreporter.ru/api/events?api-version=2>

тело запроса:

```
{
  "StartDate": "0001-01-01T00:00:00", - дата начала периода отчета
  "EndDate": "0001-01-01T00:00:00", - дата окончания периода отчета
  "IdKolonka": null, - Фильтр по ID колонки, null – все доступные
  "FirmName": null, - Фильтр по Фирме(группе), null – все доступные
  "NumPetrol": 0 - начиная с запрошенного(включительно) номера 0-все
  "FilterSystemEvent": 0 - (0-заправки, 1-системные события, 2- все)
}
```

Параметры фильтра кроме **StartDate** и **EndDate** не являются обязательными для указания, в примере указаны значения по умолчанию если они не заданы.

## Структура ответа на запрос:

```
{
  "Id": 843342, - уникальный номер записи
  "IdKolonka": "867793030485256",
  "NumPetrol": 1202,
  "IdUser": "0",
  "NameUser": "Заправка отменена получателем",
  "Firm": "Система",
  "IdTc": "0",
  "NumTc": "-",
  "Odometr": 0,
  "Waybill": 0,
  "DateTime": "2020-02-18T11:22:46",
  "AmountFuel": 0.0,
  "NameKolonka": "867793030485256",
  "Login": "7806529882",
  "UserOmnicomId": 0,
  "UnixTimeStamp": 1582014166,
  "UnixTimeStampEnd": 1582014166,
  "DateTimeEnd": "2020-02-18T11:22:46",
  "NameTank": "ТЕСТОВЫЙ", - название резервуара или секции
  "NumPost": 1, - номер поста (1-2)
  "StartTankLevel": 0.0, - уровень в резервуаре на начало события
  "EndTankLevel": 0.0, - уровень в резервуаре на конец события
  "Density": 0.0, - плотность только при подключенном уровнемере ПМП-201
  "AmountMass": 0.0, - масса (пока не используется)
  "Coast": 0.00,
  "Section": 0, - номер секции (резервуара)
  "PetrolOption": 0, - опции заправки (пока не используется)
  "Longitude": 0.0, - Долгота (координаты)
  "Latitude": 0.0, - Широта (координаты)
  "Comment": null
}
```

## Колонки (ТРК).

### Запрос списка колонок:

GET <https://garvekswebreporter.ru/api/station> вернет список доступных колонок и их статус.

Структура элемента списка колонок формат JSON:

```
{
  "IdKolodka": "012497008333381",           - уникальный идентификатор колонки
  "NameKolodka": "012497008333381",         - имя колонки
  "State": "Свободна",                       - текущий статус колонки*
  "CurLevelTank": 1000,                     - текущий уровень топлива в резервуаре**
  "OnLine": "2017-10-16T09:54:29.733",      - время последнего обмена колонки с сервером
  "SyncState": "Не подключена",             - статус связи с колонкой
  "TypeStation": "LAN"                       - интерфейс связи с колонкой
}
```

Примечание:

\* - Возможные статусы колонки:

"Свободна";	- Колонка свободна.
"Заправка";	- В данный момент происходит заправка.
"Калибровка";	- Производится калибровка счетчика.
"Остановка";	- Остановка после заправки и сохранение события в памяти колонки.
"Ошибка";	- Неисправность счетчика. Только для двухканального счетчика.
"Заблокирована";	- Колонка заблокирована диспетчером.
"Ожидание";	- Диспетчер разрешил заправку N литров, ожидание действия водителя.
"Неисправность";	- Неисправность счетчика, насоса или три нулевых заправки подряд.
"Не определен";	- Статус не может быть определен.

\*\* - Расчетный уровень топлива в резервуаре на основании предыдущих заправок.

## Пользователи.

### Получение списка пользователей:

GET <http://garvekswebreporter.ru/api/users>

вернет всех пользователей записанных в базу и их статус:

```
[
  {
    "IdUser": "1234",           - Id Карты пользователя
    "NameUser": "1 1",        - Имя(Марка ТС)\Фамилия(Модель ТС) пользователя
    "FirmName": "1",          - Название организации (Номер ТС)
    "SyncState": "Синхронизирован" - Статус пользователя
  },
  {
    "IdUser": "4567",
    "NameUser": "2345 2345",
    "FirmName": "1",
    "SyncState": "Синхронизирован"
  }
]
```

### Получение подробной информации о пользователе:

GET <http://garvekswebreporter.ru/api/users/1234>

```
[
  {
    "IdUser": "1234",
    "NameUser": "1 1",
    "FirmName": "1",
    "Password": "1",
    "ManAuth": 1,
    "KeyAuth": 1,
    "AuthBoth": 1,
    "AskNumTc": 1,
    "AskOdometr": 1,
    "AskBill": 0,
    "SyncState": "Синхронизирован"
  }
]
```

### Запись нового пользователя на колонку\колонки:

POST <https://garvekswebreporter.ru/api/users>

С параметрами в теле запроса:

```
{
  " IdUser ": "1234",           – номер карты пользователя (длина 6 )
  "userName": "1234",         – Имя пользователя /Марка ТС (длина 17)
  "userFamily": "1234",      – Фамилия пользователя / Номер ТС (длина 17)
  "userFirmName": "1234",    – Название Организации (длина 17)*
  "userPassword": "1234",   – Пароль пользователя (длина 6)
  "authMan": 0,              – Ручная авторизация (0\1)
  "authKey": 0,              – Авторизация электронным ключом (0\1)
  "authBoth": 0,            – Авторизация электронным ключом с запросом пароля (0\1)
  "askNumTc": 1,            – Запросить ввод номера ТС (0\1)
  "askOdometr": 1,          – Запросить ввод показаний одометра (0\1)
  "askWayBill": 1,          – Запросить ввод номера путевого листа (0\1)
  "station": [
    "03223223232",
    "03223223231"
  ]
}
```

**Внимание!!! Параметры авторизации одинаковые на всех колонках для одного пользователя.**

\* В программе AZS-ONLINE поле «Фирма» хранится в отдельной таблице базы, поэтому перед созданием пользователей необходимо заполнить эту таблицу.

Если организация или номер ТС не внесен в эту таблицу, при создании пользователя вы получите ошибку – «Фирма отсутствует в базе».

### Возможные ошибки при добавлении, редактировании, удалении пользователя:

```
{
  "error": [
    "Неверный ID карты",
    "ID карты уже занят",
    "Неверное имя пользователя",
    "Неверная фамилия пользователя",
    "Неверное название фирмы",
    "Фирма отсутствует в базе",
    "Неверный пароль пользователя",
    "Неверно указан ID колонки"
  ]
}
```

### Добавление пользователя на колонку/ колонки:

PUT <https://garvekswebreporter.ru/api/users>

```
{
  " IdUser ": "1234",
  "station": [
    "03223223232",
    "03223223231"
  ]
}
```

```
}
```

#### Удаление пользователя с колонки (колонок):

DELETE <https://garvekswebreporter.ru/api/users>

```
{  
  "IdUser": "1234",  
  "station": [  
    "03223223232",  
    "03223223231"  
  ]  
}
```

#### Редактирование пользователя и изменение опций авторизации:

PATCH <https://garvekswebreporter.ru/api/users>

```
{  
  "IdUser": "1234",  
  "userName": "1234",  
  "userFamily": "1234",  
  "userFirmName": "1234",  
  "userPassword": "1234",  
  "authMan": 0,  
  "authKey": 1,  
  "authBoth": 0,  
  "askNumTc": 1,  
  "askOdometr": 1,  
  "askWayBill": 1  
}
```

- номер карты редактируемого пользователя (длина 6)
- Имя пользователя / Марка ТС (длина 17)
- Фамилия пользователя / Номер ТС (длина 17)
- Название Организации (длина 17)
- Пароль пользователя (длина 6)
- Ручная авторизация (0\1)
- Авторизация электронным ключом (0\1)
- Авторизация электронным ключом с запросом пароля (0\1)
- Запросить ввод номера ТС (0\1)
- Запросить ввод показаний одометра (0\1)
- Запросить ввод номера путевого листа (0\1)

**IdUser не изменяется и не редактируется!**

**Редактирование вызовет обновление на всех колонках к которым добавлен пользователь!**

**Лимит будет перевыставлен на доступный остаток автоматически!**

## Лимиты.

### Получение лимитов пользователей:

GET <https://garvekswebreporter.ru/api/limits> вернет лимиты всех пользователей на всех колонках в виде списка со структурой указанной ниже.

### Для указанного пользователя:

GET <https://garvekswebreporter.ru/api/limits/7439>

Вернет лимиты указанного пользователя на всех колонках, на которые он записан.

```
{
  "idUser": "7439",
  "idKolonka": "012497007446789",
  "NameKolonka": "Колонка1",
  "NameTypeLimit": "На неопределенный период",
  "TypeLimit": 5,
  "StratLimit": "2017-10-09T00:00:00",
  "EndLimit": null,
  "SetLimit": 1000,
  "AvailableLimit": 1000,
  "BalanceLimit": 0,
  "SyncState": "Синхронизирован"
}
```

- ID пользователя
- ID колонки
- Имя колонки
- Наименование типа лимита
- Тип лимита
- Начало лимита
- Окончание лимита
- Заданный лимит
- Остаток лимита
- Выдано в текущем лимите
- Статус пользователя

### Установка лимита пользователю на указанной колонке:

POST <https://garvekswebreporter.ru/api/limits>

```
{
  "idUser": "6789",
  "idKolonka": "353451041297993",
  "TypeLimit": 1,
  "StratLimit": "2017-10-11T00:00",
  "EndLimit": "2017-10-11T23:59",
  "SetLimit": 340
}
```

- ID карты пользователя
- ID колонки, на которой нужно выставить лимит
- Тип лимита
- Дата начала лимита (только для типов 1 и 4)
- Дата окончания лимита (только для типов 1 и 5)
- Лимит топлива в литрах

Возможные типы лимитов:

0 – безлимит

1 – лимит на день

4 – лимит на неопределенный период

5 – лимит на определенный период (Или лимит на фирму, если он задан для фирмы, устанавливается через api фирм)

### Возможные ошибки при установке лимита:

```
{
  "error": [
    "Неверный ID карты",
    "Неверно указан ID колонки",
    "Пользователь не записан на колонку",
    "Некорректная дата начала лимита",
    "Некорректная дата окончания лимита",
    "Некорректное значение лимита"
  ]
}
```

```
}
```

## Фирмы.

### Получение списка фирм:

GET <https://garvekswebreporter.ru/api/Firm>

Вернет список фирм записанных в базу и их лимит:

```
{
  "FirmName": "1",
  "Limit": 300          - значение лимита, null – лимит не установлен
}
```

### Запись новой фирмы в базу:

POST <https://garvekswebreporter.ru/api/Firm>

```
{
  "firmName": "1"
}
```

\* - По умолчанию фирма создается с безлимитом .

### Установка лимита фирме:

PUT <https://garvekswebreporter.ru/api/Firm>

```
{
  "firmName": "1",
  "limit": 300
  "typeLimit": 0
}
```

«typeLimit» : 0 – безлимит, 1 – ограниченный лимит

### Удаление фирмы из базы:

DELETE <https://garvekswebreporter.ru/api/Firm>

```
{
  "firmName": "1"
}
```

\* - Приведет к удалению всех связанных с фирмой пользователей с колонок и из базы.

### Изменение названия фирмы:

PATCH <https://garvekswebreporter.ru/api/Firm>

```
{
  "firmName": "1",
  "newFirmName": "Тест"
}
```

## Резервуары.

### Получение списка резервуаров:

GET <https://garvekswebreporter.ru/api/StorageTank>

```
{
  "Id": 39,
  "NameTank": "Резервуар2",
  "VolumeTank": 10500,
  "WarningTank": 1000,
  "BlokedTank": 500,
  "CurrentLevel": 2387.7599999999993,
  "Login": "7811655677",
  "SendWarnSms": false,
  "SendWarnEmail": true,
  "SendBlokSms": false,
  "SendBlokEmail": true,
  "NotificationsMail": "yud06@mail.ru",
  "NotificationsSms": "",
  "LevelMeter": 0,
  "OmnicomId": 0,
  "LevelGaugeEnable": false,
  "OmnicomAccount": "yu0667@mail.ru",
  "RequireProvider": false
}
```

Уникальный ID резервуара  
Название резервуара  
объем резервуара  
уровень предупреждения резервуара  
уровень блокировки выдачи топлива  
текущий расчетный уровень  
идентификатор личного кабинета  
отправка смс при достижении уровня предупреждения  
отправка email при достижении уровня предупреждения  
отправка смс при достижении уровня блокировки  
отправка email при достижении уровня блокировки  
почта для уведомлений  
номер для sms уведомлений  
последний зафиксированный уровень уровнемером  
идентификатор резервуара в Omnicomm системе (0 -не привязан)  
Omnicom контролирует уровень в резервуаре  
Аккаунт Omnicomm привязанный к резервуару  
Требовать указание поставщика при пополнении

### Пополнение резервуара:

POST <https://garvekswebreporter.ru/api/StorageTank>

```
{
  "TankId": 39,
  Value: 200.23,
  Comment: "yu0667@mail.ru",
  IdProvider: 34
}
```

ID резервуара  
Значение пополнения в литрах  
Комментарий пополнения обязательно не более 20 знаков  
ID поставщика не обязательный, если RequireProvider : false

### Корректировка уровня резервуара:

PUT <https://garvekswebreporter.ru/api/StorageTank>

```
{
  "TankId": 39,
  Value: 2000.23
}
```

ID резервуара  
Новый уровень в литрах

**Значения при пополнении и корректировке уровня не должны выходить за заданные параметры резервуара!**

## Журнал пополнения резервуаров.

### Пополнения резервуаров за текущий день:

GET <https://garvekswebreporter.ru/api/StorageTankLog>

### Пополнения резервуара за указанный день:

GET <https://garvekswebreporter.ru/api/StorageTankLog?date=2018-03-03&tankId=39>

tankId - ID резервуара  
date - Запрашиваемый день

### Пополнения выбранного резервуара за указанный период:

POST <https://garvekswebreporter.ru/api/StorageTankLog>

```
{
  "TankId": 39,                ID резервуара
  "StartDate": "2018-01-24T08:53",  Дата и время начала отчета
  "EndDate": "2018-02-24T08:53"    Дата и время окончания отчета
}
```

### Ответ:

```
{
  "Id": 21497,                ID записи
  "Data": "2018-01-24T09:53:00.923",  Дата пополнения (корректировки) резервуара
  "Who": "Демо",              Пользователь пополнивший резервуар
  "Value": 100,               Значение корректировки/ пополнения
  "IdFirm": "7811329677",      идентификатор личного кабинета
  "TankId": 39,               ID резервуара
  "TankName": "Резервуар",     Название резервуара
  "Comments": "",             Комментарий при пополнении
  "PrevLevel": null,          Уровень перед пополнением/корректировкой
  "IdKolodka": "",           ID колонки с которой пополнили резервуар
  "NameProvider": null,       Название поставщика ГСМ
  "IdProvider": 0,            ID поставщика ГСМ
}
```

### Начальный и конечный уровень выбранного резервуара за указанный период:

PUT <https://garvekswebreporter.ru/api/StorageTankLog>

```
{
  "TankId": 39,                ID резервуара
  "StartDate": "2018-01-24T08:53",  Дата и время выборки начального уровня
  "EndDate": "2018-02-24T23:53"    Дата и время выборки конечного уровня
}
```

### Ответ:

```
{
  "TankId": 39,                ID резервуара
  "TankName": "Резервуар",     Название резервуара
  "StartLevel": 612.46,         Уровень на начальную дату выборки
  "EndLevel": 2499.68           Уровень на конечную дату выборки
}
```

## Поставщики ГСМ.

### Получение списка поставщиков:

GET <https://garvekswebreporter.ru/api/FuelProvider>

```
{
  "Id": 1,                - Внутренний ID (не используется)
  "NameProvider": "Тестовый", - Имя поставщика (длинна 20) уникальное
  "TypeFuel": "Дизель",   - Вид ГСМ (длинна 20)
  "IdFirm": "78063439882", - ID Личного кабинета админстратора
  "IdProvider": 23        - ID поставщика
}
```

### Добавление нового поставщика:

POST <https://garvekswebreporter.ru/api/FuelProvider>

```
{
  "NameProvider": "Тестовый", - Имя поставщика (длинна 20) уникальное
  "TypeFuel": "Дизель",      - Вид ГСМ (длинна 20)
  "IdProvider": 23 *         - ID поставщика (уникальный номер int)
}
```

\*- ID поставщика не может быть «0»!

### Редактирование поставщика:

PUT <https://garvekswebreporter.ru/api/FuelProvider>

```
{
  "NameProvider": "Тестовый", - Имя поставщика (длинна 20) уникальное
  "TypeFuel": "Дизель",      - Вид ГСМ (длинна 20)
  "IdProvider": 23           - ID редактируемого поставщика (Id не редактируется)
}
```

### Удаление поставщика:

DELETE <https://garvekswebreporter.ru/api/FuelProvider>

```
{
  "IdProvider": 23          - ID удаляемого поставщика
}
```